Submitted to *Econometrica*

AN AUTOMATIC FINITE-SAMPLE ROBUSTNESS METRIC: WHEN CAN DROPPING A LITTLE DATA MAKE A BIG DIFFERENCE?

T. BRODERICK
EECS, MIT

R. GIORDANO [1]
EECS, MIT

R. MEAGER [1]
Economics, LSE

We propose a method to assess the sensitivity of econometric analyses to the removal of a small fraction of the data. Manually checking the influence of all possible small subsets is computationally infeasible, so we provide an approximation to find the most influential subset. Our metric, the "Approximate Maximum Influence Perturbation," is automatically computable for common methods including (but not limited to) OLS, IV, MLE, GMM, and variational Bayes. We provide finite-sample error bounds on approximation performance. At minimal extra cost, we provide an exact finite-sample lower bound on sensitivity. We find that sensitivity is driven by a signal-to-noise ratio in the inference problem, is not reflected in standard errors, does not disappear asymptotically, and is not due to misspecification. While some empirical applications are robust, results of several economics papers can be overturned by removing less than 1% of the sample.

KEYWORDS: Robustness, Influence function, Local robustness, Z-estimators, Sensitivity.

[1]Equal contribution first authors.

Tamara Broderick and Ryan Giordano were supported in part by an Office of Naval Research Early Career Grant, an NSF CAREER Award, and an Army Research Office YIP Award. We thank Avi Feller, Jesse Shapiro, Emily Oster, Michael Kremer, Peter Hull, Tetsuya Kaji, Heather Sarsons, Kirill Borusyak, Tin Danh Nguyen and the authors of all of our applications for their insightful comments and suggestions. All mistakes are our own.

Corresponding Author: Rachael Meager, reachable at r.meager@lse.ac.uk.

1. INTRODUCTION

Ideally, policymakers will use economics research to inform decisions that affect people's livelihoods, health, and well-being. Yet study samples may differ from the target populations of these decisions in non-random ways, perhaps because of practical challenges in obtaining truly random samples, or because populations generally differ across time and place. When these deviations from the ideal random sampling exercise are small, one might think that the empirical conclusions would still hold in the populations affected by policy. It therefore seems prudent to ask whether a small percentage of a study's sample—or a handful of data points—has been instrumental in determining its findings. In this paper we provide a finite-sample, automatically-computable metric of how dropping a small amount of data can change empirical conclusions. We show that certain empirical results from high-profile studies in economics can be reversed by removing less than 1% of the sample even when standard errors are small, and we investigate why.

There are several reasons to care about whether empirical conclusions are substantially influenced by small percentages of the finite sample. In practice, even if we can sample from the population of direct interest, small percentages of the data are missing; either surveyors and implementers cannot find these individuals, or they refuse to answer our questions, or their answers get lost or garbled during data processing. As this missingness cannot safely be assumed random, researchers might care whether their substantive conclusions could conceivably be overturned by a missing handful of data points. Similarly, consumers of research who are concerned about potentially non-random errors in sample construction at any stage of the analysis might be interested in this metric as a measure of the exposure of a study's conclusions to this concern. Conclusions that are highly influenced by a small handful of data points are more exposed to adverse events or errors during data analysis, including p-hacking, even if these errors are unintentional.

Yet even if researchers could construct a perfectly random sample from a given study population, the target population for our policy decisions is always different from the study population, if only because the world may change in the time between the research and the decision. For this reason, social scientists often aspire to uncover generalizable truths about

the world and to make policy recommendations that would apply more broadly than to a single study population.

In this paper, we propose to directly measure the extent to which a small fraction of a data sample has influenced the central claims or conclusions of a study. For a particular fraction α (e.g., $\alpha = 0.001$), we propose to find the set of no more than $100\alpha\%$ of all the observations that effects the greatest change in an estimator when those observations are removed from the sample, and to report this change. For example, suppose we were to find a statistically-significant average increase in household consumption after implementing some economic policy intervention. Further suppose that, by dropping 0.1% of the sample (often fewer than 10 data points), we instead find a statistically-significant average *decrease* in consumption. Then it would be challenging to argue that there is strong evidence that this intervention would yield consumption increases in even slightly different populations.

To quantify this sensitivity, one could consider every possible $1 - \alpha$ fraction of the data, and re-run the original analysis on all of these data subsets. But this direct implementation is computationally prohibitive.[2] We propose a fast approximation that works for common estimators—including Generalized Methods of Moments (GMM), Ordinary Least Squares (OLS), Instrumental Variables (IV), Maximum Likelihood Estimators (MLE), Variational Bayes (VB), and all minimizers of smooth empirical loss. Roughly, we give each data point a weight and apply a Taylor expansion in the weights (Section 2.1 and Section 2.2). Our approximation is fast, automatable, and easy to use, and we provide an R package on GitHub called "zaminfluence."[3]

We show that our approximation performs well using theoretical analyses, simulation studies, and applied examples. We demonstrate that the approximation error is low when the percentage of the sample removed is small (Section 3.3). Moreover, for the cost of a

[2]Indeed, Young (2019) finds it computationally prohibitive to re-run their analysis when leaving out every possible subset of two data points. To illustrate, consider an analysis that takes 1 second to run; checking removal of every 4 data points from a data set of size 400 would take over 33 years. See Section 2 for more detail.

[3]https://github.com/rgiordan/zaminfluence. The name stands for "Z-estimator approximate maximum influence."

single additional data analysis, we can provide an exact lower bound on the worst-case change in an analysis upon removing $100\alpha\%$ of the data (Section 2.2.1). We check that our metric detects combinations of data points that reverse empirical conclusions when removed from real-life datasets (Section 4). For example, in the Oregon Medicaid study (Finkelstein et al., 2012), we can identify a subset containing less than 1% of the original data that controls the sign of the effects of Medicaid on certain health outcomes. In the Mexico microcredit study (Angelucci et al., 2015), we find a single observation, out of 16,500, that controls the sign of the ATE on household profit.

We investigate the source of this sensitivity when it arises, and we show that it is not captured in conventional standard errors. We find that a result's exposure to the influence of a small fraction of the sample need not reflect a model misspecification problem nor the presence of gross outliers. Sensitivity according to our metric can arise, even if the model is exactly correct and the data set large, if there is a low *signal-to-noise ratio*: that is, if the strength of the claim (signal) is small relative to a quantity which consistently estimates the standard deviation of the limiting distribution of root-N times the quantity of interest (Section 3). For example, in OLS this "noise" is large when we have a high ratio of residual variance to regressor variance (Section 3.1). This noise can be large even when standard errors are small, because it does not disappear as N grows.

We examine several applications from empirical economics papers and find that the sensitivity captured by our metric varies considerably across analyses in practice. In many cases, the sign and significance of certain estimated treatment effects can be reversed by dropping less than 1% of the sample, even when the t-statistics are very large and inference is very precise; see, e.g., the Oregon Medicaid RCT (Finkelstein et al., 2012) in Section 4.1. In Section 4.2, we examine the Progresa Cash Transfers RCT (Angelucci and De Giorgi, 2009) and show that trimming outliers in the outcome data does not necessarily reduce sensitivity. In Section 4.3 we examine an extremely simple two-parameter linear regression on seven Microcredit RCTs (Meager, 2020) and, in Section 4.4, we examine a Bayesian hierarchical analysis of the same data; among other things, these final two analyses show that neither very simple nor relatively complex Bayesian models are immune to sensitivity

to dropping small fractions of the data. However, not all analyses we examine are non-robust. Certain results across the applications we examine are robust up to 5% and even 10% removal.

We recommend that researchers use our metric to complement standard errors and other robustness checks. Our goal is not to supplant other analyses, but to provide an additional tool to be incorporated into a broader ecosystem of systematic stability analysis in data science (Yu, 2013). For example, since our approximation is fundamentally local due to the Taylor expansion, practitioners may also consider global sensitivity checks such as those proposed by Leamer (1984, 1985), Sobol (2001), Saltelli (2004), or the breakdown frontiers approach of He et al. (1990), Masten and Poirier (2020). Our method is also no substitute for tailored robustness checks designed by researchers to investigate specific concerns about sensitivity of results to certain structures or assumptions. And practitioners may benefit from robustifying their analysis (Mosteller and Tukey, 1977, Hansen and Sargent, 2008, Chen et al., 2011) even if they pass our check. Our metric is also complementary to classical robustness measures, although we are able to connect our metric to these measures via the influence function. We see that gross error sensitivity is set up for designing estimators and arbitrary adversarial perturbations to the population distribution, whereas our metric is set up for assessing sensitivity to dropping a small subset of the data at hand once an analysis has been performed. We do not recommend researchers discard results which are not robust to the removal of small highly-influential subsets of the data, but rather adjust their interpretation of such results as being less generally applicable to somewhat differing populations. We do not yet recommend any specific alterations to common inferential procedures based on our metric, but we believe this direction is promising for future research.

2. A PROPOSED METRIC

Suppose we observe N data points $d_1, \ldots, d_N$. For instance, in a regression problem, the n-th data point might consist of covariates x_n and response(s) y_n, with $d_n = (x_n, y_n)$. Consider a parameter $\theta \in \mathbb{R}^P$ of interest. Typically we estimate θ via some function $\hat{\theta}$ of our data. The central claim of an empirical economics paper is typically focused on some

attribute of θ, such as the sign or significance of a particular effect or quantity. A frequentist analyst might be worried if removing some small fraction α of the data were to

- Change the sign of an effect.
- Change the significance of an effect.
- Generate a significant result of the opposite sign.

To capture these concerns, we define the following quantities:

Definition 1. Let the *Maximum Influence Perturbation* be the largest possible change induced in the quantity of interest by dropping no more than $100\alpha\%$ of the data.

We will often be interested in the set that achieves the Maximum Influence Perturbation, so we call it the *Most Influential Set*.

And we will be interested in the minimum data proportion $\alpha \in [0, 1]$ required to achieve a change of some size Δ in the quantity of interest, so we call that α the *Perturbation-Inducing Proportion*. We report `NA` if no such α exists.

In general, to compute the Maximum Influence Perturbation for some α, we would need to enumerate every data subset that drops no more than $100\alpha\%$ of the original data. And, for each such subset, we would need to re-run our entire data analysis. If m is the greatest integer smaller than 100α, then the number of such subsets is larger than $\binom{N}{m}$. For $N = 400$ and $m = 4$, $\binom{N}{m} = 1.05 * 10^9$. So computing the Maximum Influence Perturbation in even this simple case requires re-running our data analysis over 1 billion times. If each data analysis took 1 second, computing the Maximum Influence Perturbation would take over 33 years to compute. Indeed, the Maximum Influence Perturbation, Most Influential Set, and Perturbation-Inducing Proportion may all be computationally prohibitive even for relatively small analyses.

2.1. *Setup: Notation and Assumptions*

To address this computational issue, we propose to use a (fast) approximation instead. We will see, for the cost of one additional data analysis, our approximation can provide a lower bound on the exact Maximum Influence Perturbation. More generally we provide

theory and experiments to support the quality of our approximation. We provide open-source code[4] and show that our approximation is fully automatable in practice (Section 2.2).

Our approximation is akin to a Taylor expansion, so it will require certain aspects of our estimator to be differentiable. We now summarize the assumptions under which our approximation exists, and note that many common analyses satisfy these assumptions—including, but not limited to, OLS, IV, GMM, MLE, and variational Bayes. Below, in Section 3.3, we will state stricter sufficient conditions that guarantee not only the existence but also the finite-sample accuracy of the approximation.

ASSUMPTION 1: *$\hat{\theta}$ is a* Z-estimator*; that is, $\hat{\theta}$ is the solution to the following estimating equation,*[5] *where $G(\cdot, d_n) : \mathbb{R}^P \to \mathbb{R}^P$ is a twice continuously differentiable function and 0_P is the column vector of P zeros.*

$$\sum_{n=1}^{N} G(\hat{\theta}, d_n) = 0_P. \tag{1}$$

ASSUMPTION 2: *$\phi : \mathbb{R}^P \to \mathbb{R}$, which we interpret as a function that takes the full parameter θ and returns the quantity of interest from θ, is continuously differentiable.*[6]

For instance, the function that picks out the p-th effect from the vector θ, $\phi(\theta) = \theta_p$, satisfies this assumption.

To form our approximation, we introduce a vector of data weights, $\vec{w} = (w_1, \ldots, w_N)$, where w_n is the weight for the n-th data point. We recover the original data set by giving every data point a weight of 1: $\vec{w} = \vec{1} = (1, \ldots, 1)$. We can denote a subset of the original data as follows: start with $\vec{w} = \vec{1}$; then, if the data point indexed by n is left out, set $w_n = 0$. We can collect weightings corresponding to all data subsets that drop no more than $100\alpha\%$

[4] https://github.com/rgiordano/zaminfluence

[5] Sometimes Eq. 1 is associated with "M-estimators" that optimize a smooth objective function, since such M-estimators typically take the form of a Z-estimator. However, some Z-estimators, such as IV regression, do not optimize any particular empirical objective function, so the notion of Z-estimator is in fact more general.

[6] Below, we will allow for additional dependence in ϕ on data weights.

of the original data as follows:

$$W_\alpha := \{\vec{w} : \text{No more than } \lfloor \alpha N \rfloor \text{ elements of } \vec{w} \text{ are } 0 \text{ and the rest are } 1\}. \tag{2}$$

Our main idea will be to form a Taylor expansion of our quantity of interest ϕ as a function of the weights, rather than recalculate ϕ for each data subset (i.e., for each reweighting).

To that end, we first reformulate our setup, now with the weights $\vec{w}$; note that we recover the original problem (for the full data) above by setting $\vec{w} = \vec{1}$ in what follows. Let $\hat{\theta}(\vec{w})$ be our parameter estimate at the weighted data set described by $\vec{w}$. Namely, $\hat{\theta}(\vec{w})$ is the solution to the weighted estimating equation

$$\sum_{n=1}^{N} w_n G(\hat{\theta}(\vec{w}), d_n) = 0_P. \tag{3}$$

We allow that the quantity of interest ϕ may depend on $\vec{w}$ not only via the estimator θ, so we optionally write $\phi(\theta, \vec{w})$ with $\phi(\cdot, \cdot) : \mathbb{R}^P \times \mathbb{R}^N \to \mathbb{R}$. Whenever we write $\phi(\cdot)$ as a function of a single argument, we will implicitly mean $\phi(\cdot, \vec{1})$. We require that $\phi(\cdot, \cdot)$ be continuously differentiable in both its arguments. For instance, we can use $\phi(\theta, \vec{w}) = \theta_p$ to pick out the p-th component of θ. Or, to consider questions of statistical significance, we may choose $\phi(\theta, \vec{w}) = \theta_p + 1.96\sigma_p(\theta, \vec{w})$, where $\sigma_p(\theta, \vec{w})$ is an estimate of the standard error depending smoothly on θ and $\vec{w}$; this example is our motivation for allowing the more general $\vec{w}$ dependence in $\phi(\theta, \vec{w})$.

With this notation in hand, we can restate our original goal of computing the Most Influential Set as solving

$$\vec{w}^{**} := \underset{\vec{w} \in W_\alpha}{\arg\max} \left(\phi(\hat{\theta}(\vec{w}), \vec{w}) - \hat{\phi} \right). \tag{4}$$

Here we focus on positive changes in ϕ since negative changes can be found by reversing the sign of ϕ and using $-\phi$ instead. In particular, the zero indices of $\vec{w}^{**}$ correspond to the Most Influential Set: $S_\alpha := \{n : \vec{w}^{**}_n = 0\}$. And $\Psi_\alpha = \phi(\vec{w}^{**}) - \hat{\phi}$ is the Maximum Influence Perturbation. The Perturbation Inducing Proportion is the smallest α that induces a change of at least size Δ: $\alpha^*_\Delta := \inf\{\alpha : \Psi_\alpha > \Delta\}$.

2.2. *A Tractable Approximation*

Our approximation, then, centers on a first-order Taylor expansion (and thus linear approximation) in $\vec{w} \mapsto \phi(\hat{\theta}(\vec{w}), \vec{w})$ around $\vec{w} = \vec{1}$. Let $\hat{\phi} := \phi(\hat{\theta}(\vec{1}), \vec{1})$, the quantity of interest at the original dataset. Then:

$$\phi(\hat{\theta}(\vec{w}), \vec{w}) \approx \phi^{\text{lin}}(\vec{w}) := \hat{\phi} + \sum_{n=1}^{N}(w_n - 1)\psi_n, \text{ with } \psi_n := \left.\frac{\partial \phi(\hat{\theta}(\vec{w}), \vec{w})}{\partial w_n}\right|_{\vec{w}=\vec{1}}. \quad (5)$$

We can in turn approximate the Most Influential Set as follows. Let $\psi_{(n)}$ denote the order statistics of ψ_n, i.e., the ψ_n sorted from most negative to most positive. Let $\mathbb{I}(\cdot)$ denote the indicator function taking value 0 when the argument is false and 1 when true. Then

$$\vec{w}^{**} \approx \vec{w}^{*} := \underset{\vec{w}\in W_\alpha}{\arg\max}\left(\phi^{\text{lin}}(\vec{w}) - \hat{\phi}\right) = \underset{\vec{w}\in W_\alpha}{\arg\max} \sum_{n:\, w_n = 0} (-\psi_n) \Rightarrow$$

$$\phi^{\text{lin}}(\vec{w}^{*}) - \hat{\phi} = -\sum_{n=1}^{\lfloor \alpha N \rfloor} \psi_{(n)} \mathbb{I}\left(\psi_{(n)} < 0\right). \quad (6)$$

To compute $\vec{w}^{*}$ (analogous to the $\vec{w}^{**}$ that determines the exact Most Influential Set), we compute ψ_n for each n. Then we choose $\vec{w}^{*}$ to have entries equal to zero at the $\lfloor \alpha N \rfloor$ indices n where ψ_n is most negative (and to have entries equal to one elsewhere). Analogous to the Perturbation Inducing Proportion, we can find the minimum data proportion α required to achieve a change of some size Δ: i.e., such that $\phi^{\text{lin}}(\vec{w}^{*}) - \hat{\phi} > \Delta$. In particular, we iteratively remove the most negative ψ_n (and the index n) until the Δ change is achieved; if the number of removed points is M, the proportion we report is $\alpha = M/N$. Recall that finding the exact Maximum Influence Perturbation, Most Influential Set, and Perturbation-Inducing Proportion required running a data analysis more than $\binom{M}{\lfloor \alpha N \rfloor}$ times. By contrast, our approximation requires running just the single original data analysis, N additional fast calculations to compute each ψ_n, and finally a sort on the ψ_n values.

We define our approximate quantities, as detailed immediately above, as follows.

Definition 2. The *Approximate Most Influential Set* is the set $\hat{S}_\alpha$ of at most $100\alpha\%$ data indices that, when left out, induce the biggest approximate change $\phi^{\text{lin}}(\vec{w}) - \hat{\phi}$; i.e., it is the set of data indices left out by $\vec{w}^{*}$: $\hat{S}_\alpha := \{n : \vec{w}^{*}_n = 0\}$.

The *Approximate Maximum Influence Perturbation (AMIP)* $\hat{\Psi}_\alpha$ is the approximate change observed at $\vec{w}^*$: $\hat{\Psi}_\alpha := \phi^{\text{lin}}(\vec{w}^*) - \hat{\phi}$.

The *Approximate Perturbation Inducing Proportion* $\hat{\alpha}^*_\Delta$ is the smallest α needed to cause the approximate change $\phi^{\text{lin}}(\vec{w}) - \hat{\phi}$ to be greater than Δ. That is, $\hat{\alpha}^*_\Delta := \inf\{\alpha : \hat{\Psi}_\alpha > \Delta\}$. We report `NA` if no $\alpha \in [0, 1]$ can effect this change.

Below, we will sometimes emphasize that the AMIP is a sensitivity and refer to it as the *AMIP sensitivity*. We will say that an analysis is *AMIP-non-robust* if, for a particular α of interest, the AMIP is large enough to change the substantive conclusions of the analysis. Conversely, if the AMIP is not large enough, we say an analysis is *AMIP-robust*. And we generically use the AMIP acronym to describe our methodology even when calculating the Approximate Most Influential Set or Approximate Perturbation Inducing Proportion.

2.2.1. *An exact lower bound on the Maximum Influence Perturbation*

For any problem where performing estimation a second time is not prohibitively costly, we can re-run our analysis without the data points in the Approximate Most Influential Set and thereby provide a lower bound on the exact Maximum Influence Perturbation.

Formally, let $\vec{w}^{**}$ be the weight vector for the exact Most Influential Set, and let $\vec{w}^*$ be the weight vector for the Approximate Most Influential Set $\hat{S}_\alpha$. We run the estimation procedure an extra time to recover $\phi(\hat{\theta}(\vec{w}^*), \vec{w}^*)$. Then, by definition,

$$\Psi_\alpha = \phi(\hat{\theta}(\vec{w}^{**}), \vec{w}^{**}) - \hat{\phi} = \max_{\vec{w} \in W_\alpha} \left(\phi(\hat{\theta}(\vec{w}), \vec{w}) - \hat{\phi}\right) \geq \phi(\hat{\theta}(\vec{w}^*), \vec{w}^*) - \hat{\phi}.$$

Since $\phi(\hat{\theta}(\vec{w}^*), \vec{w}^*) - \hat{\phi}$ is a lower bound for Ψ_α, we can use the Approximate Most Influential Set to conclusively demonstrate non-robustness. Of course, this lower bound holds for *any* weight vector and will be most useful if the Approximate Maximum Influence Perturbation is close to the exact Maximum Influence Perturbation. In Section 3.3 below, we establish the accuracy of the approximation for small α under mild regularity conditions.

2.2.2. *Computing the influence scores*

To finish describing our approximation, it remains to detail how to compute $\psi_n = \frac{\partial \phi(\hat{\theta}(\vec{w}), \vec{w})}{\partial w_n}\Big|_{\vec{w}=\vec{1}}$ from Eq. 5. We will refer to the quantity $\frac{\partial \phi(\hat{\theta}(\vec{w}), \vec{w})}{\partial w_n}\Big|_{\vec{w}}$ as the *influence*

score of data point n for ϕ at $\vec{w}$ since, as we show in Section 3.2 below, it is the *empirical influence function* evaluated at the datapoint d_n. To compute the influence score, we first apply the chain rule:

$$\left.\frac{\partial \phi(\hat{\theta}(\vec{w}), \vec{w})}{\partial w_n}\right|_{\hat{\theta}(\vec{w}),\vec{w}} = \left.\frac{\partial \phi(\theta, \vec{w})}{\partial \theta^T}\right|_{\hat{\theta}(\vec{w}),\vec{w}} \left.\frac{\partial \hat{\theta}(\vec{w})}{\partial w_n}\right|_{\vec{w}} + \left.\frac{\partial \phi(\theta, \vec{w})}{\partial w_n}\right|_{\hat{\theta}(\vec{w}),\vec{w}}. \tag{7}$$

The derivatives of $\phi(\cdot, \cdot)$ can be calculated using automatic differentiation software such as Python's `autograd` library (Maclaurin et al., 2015, Baydin et al., 2017). And once we have $\hat{\theta}(\vec{1})$ from running the original data analysis, we can evaluate these derivatives at $\vec{w} = \vec{1}$: e.g., $\left.\frac{\partial \phi(\theta, \vec{w})}{\partial \theta^T}\right|_{\hat{\theta}(\vec{1}),\vec{w}=\vec{1}}$.

The term $\left.\frac{\partial \hat{\theta}(\vec{w})}{\partial w_n}\right|_{\vec{w}=\vec{1}}$ requires slightly more work since $\hat{\theta}(\vec{w})$ is defined implicitly. We follow standard arguments from the statistics and mathematics literatures (Krantz and Parks, 2012, Hampel, 1974) to show how to calculate it below.

Start by considering the more general setting where $\hat{\theta}(\vec{w})$ is the solution to the equation $\gamma(\hat{\theta}(\vec{w}), \vec{w}) = 0_P$. We assume $\gamma(\cdot, \vec{w})$ is continuously differentiable with full-rank Jacobian matrix; then the derivative $\left.\frac{\partial \hat{\theta}(\vec{w})}{\partial w_n}\right|_{\vec{w}}$ exists by the implicit function theorem (Krantz and Parks, 2012, Theorem 3.3.1). We can thus use the chain rule and solve for $\left.\frac{\partial \hat{\theta}(\vec{w})}{\partial w_n}\right|_{\vec{w}}$; in what follows, $0_{P\times N}$ is the $P \times N$ matrix of zeros.

$$0_{P\times N} = \left.\frac{\mathrm{d}\gamma(\hat{\theta}(\vec{w}), \vec{w})}{\mathrm{d}\vec{w}^T}\right|_{\vec{w}} = \left.\frac{\partial \gamma(\theta, \vec{w})}{\partial \theta^T}\right|_{\hat{\theta}(\vec{w}),\vec{w}} \left.\frac{\mathrm{d}\hat{\theta}(\vec{w})}{\mathrm{d}\vec{w}^T}\right|_{\vec{w}} + \left.\frac{\partial \gamma(\theta, \vec{w})}{\partial \vec{w}^T}\right|_{\hat{\theta}(\vec{w}),\vec{w}} \tag{8}$$

$$\Rightarrow \left.\frac{\mathrm{d}\hat{\theta}(\vec{w})}{\mathrm{d}\vec{w}^T}\right|_{\vec{w}} = -\left(\left.\frac{\partial \gamma(\theta, \vec{w})}{\partial \theta^T}\right|_{\hat{\theta}(\vec{w}),\vec{w}}\right)^{-1} \left.\frac{\partial \gamma(\theta, \vec{w})}{\partial \vec{w}^T}\right|_{\hat{\theta}(\vec{w}),\vec{w}}, \tag{9}$$

where we can take the inverse by our full-rank assumption.

We apply the general setting above to our special case with $\gamma(\theta, \vec{w}) = \sum_{n=1}^{N} w_n G(\theta, d_n)$ to find

$$\left.\frac{\mathrm{d}\hat{\theta}(\vec{w})}{\mathrm{d}\vec{w}^T}\right|_{\vec{w}} = -\left(\sum_{n=1}^{N} w_n \left.\frac{\partial G(\theta, d_n)}{\partial \theta^T}\right|_{\hat{\theta}(\vec{w})}\right)^{-1} \Big(G(\hat{\theta}(\vec{w}), d_1), \ldots, G(\hat{\theta}(\vec{w}), d_N)\Big), \tag{10}$$

which can again be computed with automatic differentiation software.

2.3. *Example functions of interest*

We end this section with some concrete examples of quantities of interest. Recall from the start of Section 2 that we are often interested in whether we can change the sign or significance of an estimator, or generate a significant result of the opposite sign. Recall that $\phi(\cdot)$ with only one argument is a function of θ, and $\phi(\cdot,\cdot)$ with two arguments is a function of both θ and the weights $\vec{w}$.

To form our motivating examples, suppose for the remainder of this section we are interested in the p-th component of $\hat{\theta}$, where $\hat{\theta}_p$ is positive and statistically significant. That is, let $\hat{\sigma}_p$ be an estimator of the variance of the limiting distribution of $\sqrt{N}\hat{\phi}$, and let $\hat{\theta}_p - \frac{1.96}{\sqrt{N}}\hat{\sigma}_p$ be the lower end of our confidence interval. So we assume $\hat{\theta}_p > 0$ and $\hat{\theta}_p - \frac{1.96}{\sqrt{N}}\hat{\sigma}_p > 0$. Moreover, we will write $\hat{\sigma}_p(\theta, \vec{w})$ to emphasize that standard errors are typically given as functions of θ and the weights $\vec{w}$. For example, standard errors based on the observed Fisher information matrix $\frac{1}{N}\sum_{n=1}^{N} \vec{w}_n \left.\frac{\partial G(\theta, d_n)}{\partial \theta}\right|_{\hat{\theta}(\vec{w})}$ will, in general, depend on the weights both explicitly and through $\hat{\theta}(\vec{w})$.

To make $\hat{\theta}_p$ change sign, we can take

$$\phi(\theta) = -\theta_p. \qquad \text{(Change sign)} \tag{11}$$

We use $-\theta_p$ instead of θ_p since we have defined ϕ as a function that we are trying to increase (cf. Eq. 4 and the discussion after). Increasing $\phi(\hat{\theta})$, for ϕ in Eq. 11, by an amount $\Delta = \hat{\theta}_p$ is equivalent to $\hat{\theta}_p$ changing sign from positive to negative.

To make $\hat{\theta}_p$ statistically non-significant, we wish to take the lower bound of the confidence interval to 0. To that end, we can take

$$\phi(\theta, \vec{w}) = -\left(\theta_p - \frac{1.96}{\sqrt{N}}\hat{\sigma}_p(\theta, \vec{w})\right). \qquad \text{(Change significance)} \tag{12}$$

As in the previous case, we choose Eq. 12 with a leading negative sign because we are trying to increase ϕ (cf. Eq. 4). Increasing $\phi(\hat{\theta}, \vec{w})$, for ϕ in Eq. 12, by an amount $\Delta = \hat{\theta}_p - \frac{1.96}{\sqrt{N}}\hat{\sigma}_p$ is equivalent to $\hat{\theta}_p$ becoming statistically insignificant.

Similarly, to change to a significant result of the opposite sign, we can take

$$\phi(\theta, \vec{w}) = -\left(\theta_p + \frac{1.96}{\sqrt{N}}\hat{\sigma}_p(\theta, \vec{w})\right) \qquad \text{(Significant sign reversal)}$$

and $\Delta = \hat{\theta}_p + \frac{1.96}{\sqrt{N}}\hat{\sigma}_p$, for if the upper end of the confidence interval is negative, then the estimator must be negative and statistically significant.

In each case above, the quantity Δ represents how far we must move ϕ in order to reverse our conclusions. In this sense, Δ is a measure of the amount of "signal" in the original dataset. As we will discuss in Section 3 below, the signal Δ is one of the three key quantities that determine AMIP robustness.

2.4. *A real-world OLS regression example*

Before continuing, we illustrate our method with an example. Economists often analyze causal relationships using linear regressions estimated via ordinary least squares (OLS), but a researcher rarely believes the conditional mean dependence is truly linear. Rather, researchers use linear regression since it allows transparent and straightforward estimation of an average treatment effect or local average treatment effect. Researchers often invoke the law of large numbers to justify the focus on the sample mean, and invoke the central limit theorem to justify the use of Gaussian confidence intervals when the sample is large. We now discuss an example from recent economics literature showing how, in practice, the omission of a very small number of data points can have outsize influence on regression parameters in the finite sample even when the full sample is large. We will study AMIP sensitivity for OLS further using simulation and theory in Section 3.1 below.

Consider as an example the set of seven randomized controlled trials of expanding access to microcredit discussed by Meager (2019). For illustrative purposes we single out the study with the largest sample size: Angelucci et al. (2015). This study has approximately 16,500 households. A full treatment of all seven studies is in Sections 4.3 and 4.4 along with tables and figures of the results discussed below.

We consider the headline results on household business profit regressed on an intercept and a binary variable indicating whether a household was allocated to the treatment group

or to the control group. Let Y_{ik} denote the profit measured for household i in site k, and let T_{ik} denote their treatment status. We estimate the following model via OLS with the regression formula $Y_{ik} \sim \beta_0 + \beta_1 T_{ik}$. In the notation of Section 2.1, we have $\theta = (\beta_0, \beta_1)^T$, $d_{ik} = (Y_{ik}, T_{ik})$ with $n = (i, k)$, and $G(\theta, d_{ik}) = (Y_{ik} - (\beta_0 + \beta_1 T_{ik}))(1, T_{ik})^T$.

We confirm the main findings of the study in estimating a non-significant average treatment effect (ATE) of -4.55 USD PPP per 2 weeks, with a standard error of 5.88. We are interested in whether we can change the sign of β_1 from negative to positive, so we take $\phi(\theta) = \beta_1$. We compute ψ_n for each data point in the sample, which takes only a fraction of a second in R using our Zaminfluence package.

Examining $\vec{\psi}$, we find that one household has $\psi_n = 4.95$; removing that single household should flip the sign if the approximation is accurate. We manually remove the data point and re-run the regression, and indeed find that the ATE is now 0.4 with a standard error of 3.19. Moreover, by removing 15 households we can generate an ATE of 7.03 with a standard error of 2.55: a significant result of the opposite sign.

How is it possible for the absence of a single household to flip the sign of an estimate that was ostensibly based on all the information from a sample of 16,500? It may be tempting to suspect the use of sample means, which are known to be non-robust to gross errors, or to speculate that such excess sensitivity is simply symptomatic of ordinary sampling noise which is captured adequately by standard errors. In Section 3 to follow, we show that such intuition is not correct. On the contrary, AMIP robustness is in fact fundamentally different than both standard errors and classical robustness to gross errors.

3. UNDERLYING THEORY AND INTERPRETATION

We now establish the determinants and accuracy of AMIP robustness. We begin by deriving the key quantities of AMIP robustness in the simple case of correctly specified univariate OLS regression (Section 3.1). For this simple case, we show with theory and simulations that AMIP robustness is not necessarily driven by misspecification, that AMIP non-robustness does not vanish asymptotically, and that AMIP robustness is distinct from standard errors. Next, we formally extend these conclusions to general Z-estimators in Sec-

tion 3.2. Finally, in Section 3.3, we establish conditions under which the approximation is provably uniformly accurate for small α, both in finite sample and asymptotically.

We will see that a central equation in our understanding of AMIP robustness is its decomposition into three key quantities: the signal, noise, and shape. First, the *signal* Δ is the size of change in our quantity of interest that would reverse our substantive conclusion (see Section 2.3 above). Large values of the signal Δ indicate that large changes are needed to make a different decision. Second, the *noise* $\hat{\sigma}_\psi$ is defined by

$$\hat{\sigma}_\psi^2 := \frac{1}{N}\sum_{n=1}^{N}(N\psi_n)^2 \tag{13}$$

We call $\hat{\sigma}_\psi$ the noise because $\hat{\sigma}_\psi^2$ is typically a consistent estimator of the variance of the limiting distribution of $\sqrt{N}\phi(\hat{\theta})$, a fact that will follow below from the relationship between AMIP robustness, robust standard error estimators, and the influence function (see Section 3.2.1, paragraph (b) or, more generally, Section 3.2.3, paragraph (d)). Third, the *shape* $\hat{\mathscr{T}}_\alpha$ is defined as

$$\hat{\mathscr{T}}_\alpha := -\frac{1}{N}\sum_{n=1}^{\lfloor \alpha N \rfloor}\frac{N\psi_{(n)}}{\hat{\sigma}_\psi}\mathbb{I}\left(\psi_{(n)} < 0\right), \tag{14}$$

where $\psi_{(n)}$ refers to the n-th order statistic of the influence scores, and $\mathbb{I}(\cdot)$ denotes the indicator function taking value 1 when its argument is true and 0 otherwise. The shape $\hat{\mathscr{T}}_\alpha$ depends in a complicated way on the shape of the tail of the distribution of the influence scores, but we show that $0 \leq \hat{\mathscr{T}}_\alpha \leq \sqrt{\alpha(1-\alpha)}$ with probability one, and that $\hat{\mathscr{T}}_\alpha$ converges in probability to a nonzero constant under standard assumptions (see Section 3.2.1, paragraph (c)). Given these three quantities, we will show in Section 3.2.1, paragraph (a) that

$$\text{An analysis is AMIP non-robust} \quad \Leftrightarrow \quad \frac{\Delta}{\hat{\sigma}_\psi} \leq \hat{\mathscr{T}}_\alpha. \tag{15}$$

We refer to the quantity $\Delta/\hat{\sigma}_\psi$ as the *signal-to-noise ratio*. For a given α, Eq. 15 suggests that it is the signal-to-noise ratio that primarily determines AMIP robustness. Additionally,

this decomposition allows us to succinctly compare AMIP robustness to standard errors and gross-error robustness, as well as to analyze the large-N behavior of AMIP robustness.

This section will use the following notation. Let the symbol $\xrightarrow{p}$ denote convergence in probability, and $\rightsquigarrow$ denote convergence in distribution, both as $N \to \infty$. Let $\|\cdot\|_{op}$ denote the operator norm of a matrix.

3.1. *Theory and interpretation for Ordinary Least Squares*

We begin by focusing on the simple case of correctly-specified univariate linear regression, both to provide intuition and motivate the more general results that follow.

3.1.1. *Problem setup for Ordinary Least Squares example*

(a) Model. Let $X = (x_1, \ldots, x_N)^T$ denote a vector of N continuous mean-zero regressors, drawn IID from a distribution with finite variance σ_x^2. Let $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_N)$ be a vector of IID draws from a $\mathcal{N}(0, \sigma_\varepsilon^2)$ distribution, where we will assume σ_ε is known. For some unknown $\theta_0 \in \mathbb{R}$, let $y_n = \theta_0 x_n + \varepsilon_n$, so that the vector $Y = (y_1, \ldots, y_N)$ given X is drawn from a correctly specified regression model with true coefficient θ_0.

(b) Weighted estimating equation. The OLS estimator $\hat{\theta}$ is traditionally found by maximizing the (log) likelihood: $\log p(y_n|\theta, x_n) = -\frac{1}{2\sigma_\varepsilon^2}(y_n - \theta x_n)^2 + C$, where C does not depend on θ. In particular, setting the derivative of the log likelihood to zero yields the estimating equation $G(\theta, d_n) = -\frac{1}{\sigma_\varepsilon^2}(y_n - \theta x_n)x_n = 0$. That is, $\hat{\theta}$ is a Z-estimator with this choice of G (see Eq. 1). Typical Z-estimators do not have closed-form solutions. But in this case, the solution to the estimating equation returns the usual OLS estimate. A similar derivation returns the solution to the weighted estimating equation given in Eq. 3: $\hat{\theta}(\vec{w}) = \left(\frac{1}{N}\sum_{n=1}^N \vec{w}_n x_n^2\right)^{-1} \frac{1}{N}\sum_{n=1}^N \vec{w}_n y_n x_n$.

(c) Quantity of interest. Suppose we are interested in the sign of θ_0. Without loss of generality, we assume $\hat{\theta} < 0$. Then our quantity of interest is $\phi(\theta) = \theta$.

(d) Signal and noise. For our quantity of interest, the signal is $\Delta = \left|\hat{\theta}\right|$ since, if we can increase $\hat{\theta}$ by an amount $\left|\hat{\theta}\right|$, its sign will change. To compute the noise, we compute the influence scores. Directly differentiating the explicit formula for $\hat{\theta}$ gives, as it must, the same value for ψ_n as the implicit function theorem result of Eq. 9. Letting

$\hat{\varepsilon}_n := y_n - \hat{\theta} x_n$ and $S_X := \frac{1}{N}\sum_{n=1}^N x_n^2$, we see, either by direct differentiation or by Eq. 9, that $\psi_n = N^{-1} S_X^{-1} x_n \hat{\varepsilon}_n$. For intuition about the noise $\hat{\sigma}_\psi$, we observe its asymptotic behavior. Standard results for OLS give:

$$\hat{\sigma}_\psi^2 = \frac{1}{N}\sum_{n=1}^N (N\psi_n)^2 = S_X^{-2}\frac{1}{N}\sum_{n=1}^N x_n^2 \hat{\varepsilon}_n^2 \xrightarrow{p} \frac{\sigma_\varepsilon^2}{\sigma_x^2}. \tag{16}$$

Note that the noise includes a contribution from both the residual and regressor variance—we describe $\hat{\sigma}_\psi$ as the "noise" because it estimates the variability of $\sqrt{N}\hat{\theta}$, not of the residuals (see Section 3.1.2, paragraph (e) below). Finally, we emphasize that, although we will be using asymptotics to provide intuition, by "noise" we will always mean the finite-sample quantity $\hat{\sigma}_\psi$, not its asymptotic limit.

3.1.2. *What determines AMIP robustness for Ordinary Least Squares?*

Now that we have translated OLS into our framework, we can analyze the AMIP for OLS. To that end, we use both theory and a simulation study. We outline the simulation study before describing our main conclusions. For $N = 5,000$ data points, and for a range of σ_x and σ_ε, we drew normal regressors $x_n \sim \mathcal{N}(0, \sigma_x^2)$ and residuals $\varepsilon_n \sim \mathcal{N}(0, \sigma_\varepsilon^2)$. For $\theta_0 = 0.5$, we set $y_n = \theta_0 x_n + \varepsilon_n$. We computed the OLS estimator $\hat{\theta} = \sum_{n=1}^N y_n x_n / \sum_{n=1}^N x_n^2$.

(a) Signal-to-noise ratio drives AMIP robustness. From our discussion at the start of Section 3, we expect that the signal-to-noise ratio drives whether an analysis is AMIP-robust or not. In our simulation, N is large and we keep θ_0 fixed, so we expect that the signal does not change substantially over the simulation. Therefore, signal-to-noise is controlled by the noise. Following the asymptotic argument above, we approximate the noise as $\sigma_\varepsilon/\sigma_x$. In the left panel of Figure 1, we vary σ_ε and σ_x and plot the resulting Approximate Perturbation Inducing Proportion α^* to change the sign of $\hat{\theta}$. As expected, we see that the simulations with the largest approximate noise $\sigma_\varepsilon/\sigma_x$ are the least robust, in the sense that one can reverse the sign of $\hat{\theta}$ by dropping a very small proportion of points.

(b) Influential data points have both a large residual and large regressor. Let $(\hat{\varepsilon}x)_{(n)}$ denote the products $\hat{\varepsilon}_n x_n$, sorted from most negative to most positive, so that the sorted in-

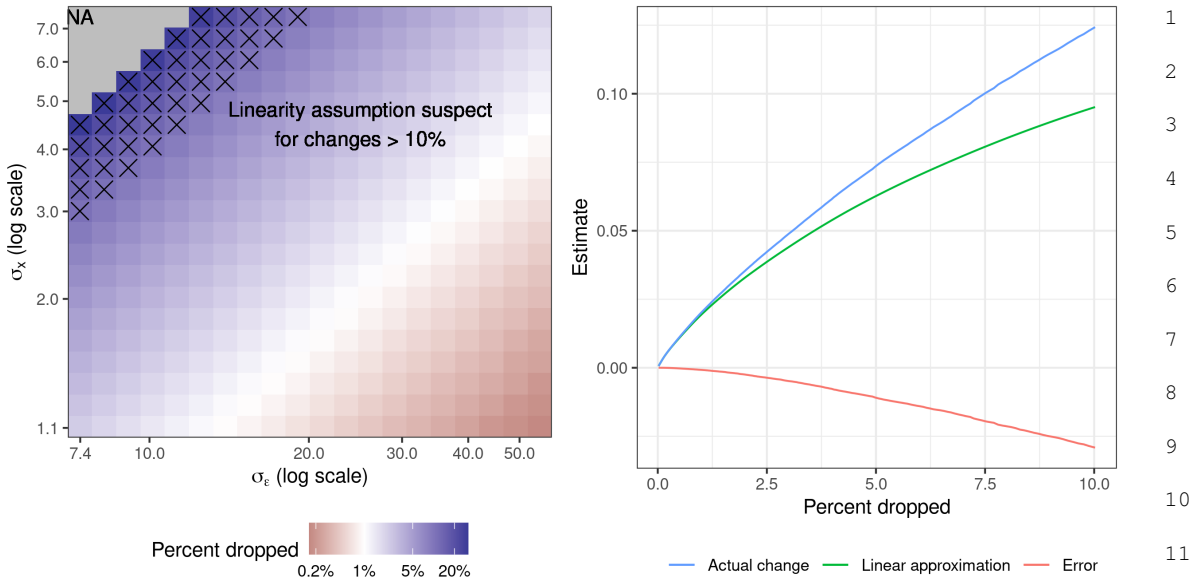


FIGURE 1.—Simulation results for univariate linear regression with $N = 5,000$ observations. **Left panel:** The approximate perturbation inducing proportion at differing values of σ_x and σ_ε. Red colors indicate datasets whose sign can is predicted to change when dropping less than 1% of datapoints. The grey areas indicate $\hat{\Psi}_\alpha = \texttt{NA}$, a failure of the linear approximation to locate any way to change the sign. **Right panel:** The actual change, linear approximation to the change, and approximation error for $\sigma_x = 2$ and $\sigma_\varepsilon = 1$.

fluence scores are $\psi_{(n)} = N^{-1} S_X^{-1} (\hat{\varepsilon} x)_{(n)}$. From this formula, we observe that influential datapoints have both a large residual and a large regressor (relative to the regressor variance).[7] A typical influence score goes to zero at rate N^{-1}, though extreme values such as $\max_n |\psi_n|$ may obey a different rate. However, since $\frac{1}{N}\sum_{n=1}^N x_n^2$ and $\frac{1}{N}\sum_{n=1}^N \varepsilon_n^2$ are finite with high probability, even $\max_n |\psi_n|$ does not diverge in this case.[8]

(c) AMIP sensitivity does not vanish as $N \to \infty$. Standard results for OLS give that $S_X \xrightarrow{p} \sigma_x^2$ and $\hat{\varepsilon}_n - \varepsilon_n \xrightarrow{p} 0$. So $N\psi_n - \sigma_x^{-2} x_n \varepsilon_n \xrightarrow{p} 0$. Consequently, the empirical distribu-

[7] Indeed, if we had taken $\phi(\theta) = \hat{\theta} x_n = \hat{y}_n$, then the n-th influence score would have been $S_X^{-1} x_n^2 \hat{\varepsilon}_n$, which is precisely the leverage score times the residual. This expression formalizes the conceptual link made by Chatterjee and Hadi (1986) between influence, leverage, and large values of $\hat{\varepsilon}_n$.

[8] The finiteness follows from the inequality $\frac{1}{N}\max_n x_n^2 \leq \frac{1}{N}\sum_{n=1}^N x_n^2 \xrightarrow{p} \sigma_x^2$, with an analogous inequality for ε_n. However, since we know ε_n is Gaussian, we actually have a stronger result in this case: $\max_{n \in \{1,\ldots,N\}} |\varepsilon_n|$ grows at rate $\sqrt{\log(2N)}$ (Rigollet, 2015, Theorem 1.14).

tion of $N\psi_n$ converges to a non-degenerate distribution with finite variance. Let q_α denote the α-th quantile of the distribution of the random variable $\sigma_x^{-2} x_1\varepsilon_1$. Since x_n and ε_n are independent, and about half of the ε_n will be negative, we expect about half of the influence scores to be negative. So for $\alpha \ll 1/2$, with high probability at least αN influence scores are negative. Then, by Eq. 6 and Slutsky's theorem, we have

$$\phi^{\text{lin}}(\vec{w}^*) - \hat{\phi} = -\sum_{n=1}^{\alpha N} \psi_{(n)} = -\frac{1}{N}\sum_{n=1}^{\alpha N} S_X^{-1}(\hat{\varepsilon}x)_{(n)} \xrightarrow{p} \mathbb{E}\left[-\frac{x_1\varepsilon_1}{\sigma_x^2}\mathbb{I}\left(\frac{x_1\varepsilon_1}{\sigma_x^2} \le q_\alpha\right)\right].$$

The right hand side of the preceding display is strictly positive for finite α. So, for fixed α, we expect that AMIP sensitivity does not vanish as $N \to \infty$.[9]

(d) AMIP non-robustness is not due only to misspecification. Our simulations are well specified. Yet we see from Figure 1 that different cases can still be robust or non-robust under various robustness cut-offs—according to their differing signal-to-noise ratios.

Asymptotically as $N \to \infty$, even in a well-specified model, we in fact expect AMIP non-robustness at any α for a sufficiently small $|\theta_0|$. The limiting value of the AMIP sensitivity does not depend on θ_0. Thus, as $N \to \infty$, our quantity of interest (for changing the sign of the estimator) will be AMIP non-robust with high probability if and only if $|\theta_0| < \mathbb{E}\left[-\frac{x_1\varepsilon_1}{\sigma_x^2}\mathbb{I}\left(\frac{x_1\varepsilon_1}{\sigma_x^2} \le q_\alpha\right)\right]$. If we are interested in the sign of θ_0, and $|\theta_0|$ is small relative to the tail means of $\sigma_X^{-2} x_1\varepsilon_1$, then the problem will be AMIP non-robust with probability approaching one, no matter how large N is—despite the fact that the model is correctly specified and there are no abnormalities in the data.

(e) Though both are scaled by the noise, standard errors are different from—and typically smaller than—AMIP sensitivity. In what may seem at first like a remarkable coincidence, the variance of the limiting distribution of $N\psi_n$ (which determines AMIP sensitivity—see Eq. 5) is the same as the variance of the limiting distribution of our quantity of interest $\sqrt{N}(\hat{\theta} - \theta_0)$ (which determines classical standard errors). The two distributions are not the same—the limiting distribution of $N\psi_n$ is not, in general, normal—but they have the same scale. In particular, compare the noise limit in Eq. 16 with the following

[9] As desired, though, the expectation does go to zero as $\alpha \to 0$ since $\mathbb{E}\left[|x_1\varepsilon_1|\right] < \infty$.

limit, which follows by standard results for OLS.

$$\sqrt{N}(\hat{\theta} - \theta_0) \rightsquigarrow \mathcal{N}\left(0, \frac{\sigma_\varepsilon^2}{\sigma_x^2}\right).$$

As we discuss below in Section 3.2.1, paragraph (b) and Section 3.2, paragraph (d), this equality is no coincidence, but a general (and well-known) relationship between influence scores and the limiting distributions of quantities of interest.

For large N, use of standard errors will admit the hypothesis that θ_0 might be 0 whenever $|\theta_0| < \frac{1.96}{\sqrt{N}} \frac{\sigma_\varepsilon}{\sigma_x}$. Thus, for every $\theta_0 \neq 0$, using standard errors always rejects $\theta_0 = 0$ for sufficiently large N. By contrast, as we saw above, using the AMIP will admit a change large enough to move $\hat{\theta}$ to 0 whenever

$$|\theta_0| \leq \left(\mathbb{E}\left[-\frac{x_1}{\sigma_x}\frac{\varepsilon_1}{\sigma_\varepsilon}\mathbb{I}\left(\frac{x_1}{\sigma_x}\frac{\varepsilon_1}{\sigma_\varepsilon} \leq \frac{\sigma_x}{\sigma_\varepsilon}q_\alpha\right)\right]\right)\frac{\sigma_\varepsilon}{\sigma_x} \neq \frac{1.96}{\sqrt{N}}\frac{\sigma_\varepsilon}{\sigma_x}.$$

Thus, we see that both the AMIP sensitivity and standard errors admit larger possible values for $\hat{\theta}$ when the limiting value $|\theta_0|/(\sigma_\varepsilon/\sigma_x)$ of the signal-to-noise ratio is large. But AMIP sensitivity is determined by the tail mean of the standardized influence scores, and standard errors are determined by a quantity that goes to zero as $N \to \infty$. Thus AMIP sensitivity is distinct from, and typically larger than, standard errors. The tail behavior of the unit-variance random variable $\frac{x_1}{\sigma_x}\frac{\varepsilon_1}{\sigma_\varepsilon}$ is exactly the shape we introduced at the start of Section 3. The shape captures the scale-independent shape of the tails of the distribution of the influence scores; see Section 3.2.1, paragraph (c) below for a detailed and general analysis.

(f) Our approximation is accurate for small α. The expression for $\hat{\theta}(\vec{w})$ depends on two terms, $\left(\frac{1}{N}\sum_{n=1}^N \vec{w}_n x_n^2\right)^{-1}$ and $\frac{1}{N}\sum_{n=1}^N \vec{w}_n y_n x_n$, both of which are uniformly smooth functions of $\vec{w}/N$ with high probability for sufficiently small $\left\|\vec{w} - \vec{1}\right\|_2/N$. As a consequence of smoothness, we expect a linear approximation formed at $\vec{w} = \vec{1}$ to be accurate when $\left\|\vec{w} - \vec{1}\right\|_2/N$ is small. And when $\vec{w}$ contains no more than $\lfloor \alpha N \rfloor$ zeros and the rest ones, we have that $\left\|\vec{w} - \vec{1}\right\|_2/N \leq \alpha$, so we expect a linear approximation to be accurate when α is small. We make this intuition precise and general in Section 3.3 below.

We check the accuracy of the approximation empirically in Figure 1. For the right hand plot in Figure 1, we fixed $\sigma_\varepsilon = 1$ and $\sigma_x = 2$. We computed the Approximate Most Influential Set for a range of left-out proportions α from 0 to 10%. For each α, we computed the linear approximation, re-ran the regression to compute the actual change, and computed the error of the linear approximation as the difference of the two. The right panel of Figure 1 shows how the relative error of the approximation vanishes for small α, and that, qualitatively, the approximation is very good for removal proportions less than 2.5%.

3.2. *Theory and interpretation for general Z-estimators*

We next show that the conclusions of Section 3.1 hold not just for OLS but in considerable generality for Z-estimators. In the present section, we will establish more generally that AMIP sensitivity is not a product of misspecification, does not vanish as N goes to infinity, and is distinct from standard errors. To that end, in Section 3.2.1 we first formally decompose the AMIP into the shape and noise terms defined at the beginning of Section 3, and we establish that the shape is roughly constant across distributions. Then, in Section 3.2.2, we use this decomposition to revisit our OLS conclusions about AMIP sensitivity but now more broadly. Finally, in Section 3.2.3, we connect the AMIP to the influence function, showing how AMIP robustness is different from gross error robustness.

3.2.1. *The decomposition of the AMIP*

(a) The AMIP is the noise times the shape. Let $\psi_{(1)}, \ldots, \psi_{(N)}$ denote the order statistics of the influence scores. Recall that the Approximate Maximum Influence Perturbation is given by the negative of the sum of the $\lfloor \alpha N \rfloor$ largest influence scores. So we can write

$$\hat{\Psi}_\alpha = \phi^{\text{lin}}(\vec{w}^*) - \hat{\phi} = -\sum_{n=1}^{\lfloor \alpha N \rfloor} \psi_{(n)} \mathbb{I}\left(\psi_{(n)} < 0\right) = \hat{\sigma}_\psi \hat{\mathscr{T}}_\alpha. \tag{17}$$

The first equality follows from the definition of the AMIP $\hat{\Psi}_\alpha$ (Definition 2). The second equality follows from Eq. 6. The third equality follows from the definitions of noise $\hat{\sigma}_\psi$ and shape $\hat{\mathscr{T}}_\alpha$ at the start of Section 3.

(b) The noise is an estimator of the standard deviation of the limiting distribution of the quantity of interest (Z-estimator version). In the case of Z-estimators, we can show by direct computation that $\hat{\sigma}_\psi^2$ is the estimator of the variance of the limiting distribution of $\sqrt{N}\phi(\hat{\theta})$ given by the delta method and the "sandwich" or "robust" covariance estimator (Huber, 1967, Stefanski and Boos, 2002). To see this, observe first that $\frac{1}{N}\sum_{n=1}^{N} \frac{\mathrm{d}\hat{\theta}(\vec{w})}{\mathrm{d}\vec{w}_n}\Big|_{\vec{1}} \left(\frac{\mathrm{d}\hat{\theta}(\vec{w})}{\mathrm{d}\vec{w}_n}\Big|_{\vec{1}}\right)^T$, as given by Eq. 10, is precisely the sandwich covariance estimator for the covariance of the limiting distribution of $\sqrt{N}\hat{\theta}$. In turn, the sample variance of the linear approximation given in Eq. 7, given by $\hat{\sigma}_\psi^2$, is then the delta method variance estimator for $\sqrt{N}\hat{\phi}$. Note that we came to the same conclusion in the special case of OLS in Section 3.1.2, paragraph (e) above.

It follows that we can use $\hat{\sigma}_\psi$ to form consistent credible intervals for ϕ, a fact that will be useful below when comparing AMIP robustness to standard errors. Specifically, if $\hat{\sigma}_\psi \xrightarrow{p} \sigma_\psi$ and $\hat{\theta} \xrightarrow{p} \theta_\infty$, then

$$\sqrt{N}(\phi(\hat{\theta}) - \phi(\theta_\infty)) \rightsquigarrow \mathcal{N}(0, \sigma_\psi^2). \tag{18}$$

As we discuss in Section 3.2.3, paragraph (d) below, this relationship between asymptotic variance and the influence scores is in fact a consequence of a general relationship between influence functions and distributional limits.

(c) The shape depends primarily on α, not on the model specification. More precisely, we next show that the shape $\hat{\mathscr{T}}_\alpha$ satisfies the following properties. (1) With probability one, $0 \leq \hat{\mathscr{T}}_\alpha \leq \sqrt{\alpha(1-\alpha)}$. (2) Typically, $\hat{\mathscr{T}}_\alpha$ converges in probability to a nonzero constant as $N \to \infty$. (3) $\hat{\mathscr{T}}_\alpha$ is largest when the the influence scores of the left-out points are all equal. Conversely, heavy tails in the distribution of ψ_n result in smaller values of $\hat{\mathscr{T}}_\alpha$. (4) Empirically, $\hat{\mathscr{T}}_\alpha$ varies relatively little among common sampling distributions.

To prove the lower bound in (1), we observe that the indicator $\mathbb{I}\left(\psi_{(n)} < 0\right)$ accounts for the fact that the adversarial weight would leave out fewer points rather than drop a point

with positive $\psi_{(n)}$. Because of this, $\hat{\mathscr{T}}_\alpha \geq 0$. We show the upper bound of (1) as part of the extremization argument for (3) below.

To prove (2), notice that $\hat{\mathscr{T}}_\alpha$ is a sum of $\lfloor \alpha N \rfloor$ positive terms, divided by N. In general, then, we expect $\hat{\mathscr{T}}_\alpha$ to converge to a nonzero constant for fixed α as long as the distribution of $N\psi_n$ converges marginally in distribution to a non-degenerate random variable. And indeed, by Eqs. 7 and 10, we expect such convergence from Slutsky's theorem as long as $\hat{\theta}$ and $\frac{1}{N}\sum_{n=1}^{N} \frac{\partial G(\hat{\theta}, d_n)}{\partial \theta}\Big|_{\hat{\theta}}$ converge in probability to constants, since $N\psi_n$ is proportional to $G(\hat{\theta}, d_n)$, which itself has a non-degenerate limiting distribution.

We next show (3), that $\hat{\mathscr{T}}_\alpha$ takes its largest possible value when all the influence scores $\psi_{(1)}, \ldots, \psi_{(\alpha N)}$ take the same negative value. To that end, take αN to be an integer for simplicity. By the definition of $\hat{\sigma}_\psi$ (Eq. 13), $\frac{1}{N}\sum_{n=1}^{N}\left(\frac{N\psi_{(n)}}{\hat{\sigma}_\psi}\right)^2 = 1$, and by properties of the influence function detailed below, $\sum_{n=1}^{N}\psi_n = 0$ (Section 3.2.3, paragraph (c)). So $\hat{\mathscr{T}}_\alpha$ is a tail average of scalars with zero sample mean and unit sample variance. Therefore, it is equivalent to consider scalars $z_1, \ldots, z_N$ with $\frac{1}{N}\sum_{n=1}^{N} z_n = 0$ and $\frac{1}{N}\sum_{n=1}^{N} z_n^2 = 1$ and to ask how to maximize the average $-\frac{1}{\alpha N}\sum_{n=1}^{N\alpha} z_{(n)}$.

To perform this maximization we divide datapoints into a set D of dropped indices, and set K of kept indices. To be precise, $D := \{n : z_{(n)} \leq z_{(\alpha N)}\}$ and $K := \{1, \ldots, N\} \setminus D$. We write the sample means and variances within the sets respectively as $\mu_D := \frac{1}{\alpha N}\sum_{n\in D} z_n$ and $v_D := \frac{1}{\alpha N}\sum_{n\in D}(z_n - \mu_D)^2$, with analogous expressions for μ_K and v_K. In this notation, our goal is to extremize μ_D, the mean in the dropped set. The constraints on the distribution can then be written as $\frac{1}{N}\sum_{n=1}^{N} z_n = 0 \Rightarrow \alpha\mu_D + (1-\alpha)\mu_K = 0$, and $\frac{1}{N}\sum_{n=1}^{N} z_n^2 = 1 \Rightarrow \alpha(v_D + \mu_D^2) + (1-\alpha)(v_K + \mu_K^2) = 1$. Given these constraints, we extremize μ_D by setting $v_K = v_D = 0$, in which case we achieve $\mu_D = -\sqrt{(1-\alpha)/\alpha}$. Identifying $N\psi_n/\hat{\sigma}_\psi$ with z_n, and $\hat{\mathscr{T}}_\alpha$ with $\alpha\mu_D$, we see that the worst-case value of $\hat{\mathscr{T}}_\alpha$ occurs when all the influence scores $\psi_{(1)}, \ldots, \psi_{(\alpha N)}$ take the same negative value. This observation completes our argument for (3). It also follows from this argument that $\hat{\mathscr{T}}_\alpha \leq \sqrt{\alpha(1-\alpha)}$ with probability one, a bound that is achieved in the worst-case. This observation supplies the upper bound in (1).

To establish point (4), we fix a representative α, simulate a large number of IID draws $\tilde{z}_n$ from some common distributions, standardize to get $z_n := \frac{\tilde{z}_n - \bar{\tilde{z}}}{\sqrt{\frac{1}{N}\sum_{n=1}^{N}(\tilde{z}_n - \bar{\tilde{z}})^2}}$, and compute the shape $\hat{\mathscr{T}}_\alpha = -\frac{1}{N}\sum_{n=1}^{\lfloor \alpha N \rfloor} z_{(n)}$. We find that, across common distributions, $\hat{\mathscr{T}}_\alpha$ varies relatively little. For example, for $\alpha = 0.01$, a Normal distribution gives $\hat{\mathscr{T}}_\alpha = 0.0266$, a Cauchy distribution gives $\hat{\mathscr{T}}_\alpha = 0.0022$. As expected based on the reasoning of the previous paragraph, the heavy-tailed Cauchy distribution has a smaller shape than the Normal distribution. The worst-case distribution, for which all left-out z_n are equal, gives $\hat{\mathscr{T}}_\alpha = 0.0995 \approx \sqrt{\alpha(1-\alpha)}$ as expected.

3.2.2. *What determines AMIP robustness?*

We now use the decomposition of the AMIP into noise and shape, and the relative stability of the shape, to derive a number of general properties of AMIP robustness.

(a) Signal-to-noise ratio drives AMIP robustness. We argued above that we do not expect $\hat{\mathscr{T}}_\alpha$ to vary radically from one problem to another. By contrast, the noise $\hat{\sigma}_\psi$ can, in principle, be any positive number. We conclude then, that the signal-to-noise ratio, rather than the shape, principally determines AMIP robustness.

This relationship also suggests what might be done if the analysis is deemed AMIP non-robust. Since, as we showed in Section 3.2.1, paragraph (b), $\hat{\sigma}_\psi$ is thus the same quantity that enters standard error computations, analysts are typically attentive to choosing estimators with $\hat{\sigma}_\psi$ as small as possible while still guaranteeing desirable properties like consistency. Meanwhile, the signal Δ is determined by the question being asked and the true state of nature as estimated by $\hat{\theta}$. In light of these observations, consider a case where $\Delta/\hat{\sigma}_\psi$ is too small to ensure AMIP robustness. Then it seems necessary for the investigator to ask a different question, or investigate different data, to find an AMIP robust analysis.

(b) AMIP sensitivity does not vanish as $N \to \infty$. Both $\hat{\sigma}_\psi$ and $\hat{\mathscr{T}}_\alpha$ converge to nonzero constants. So $\hat{\sigma}_\psi \hat{\mathscr{T}}_\alpha$, the estimated amount by which you can change an estimator, does not go to zero, either. If the signal Δ is less than the probability limit of $\hat{\sigma}_\psi \hat{\mathscr{T}}_\alpha$, then the problem

will be AMIP non-robust no matter how large N grows. As we discuss below, this behavior contrasts sharply with the behavior of standard errors.

(c) AMIP non-robustness is not due only to misspecification. Consider a correctly-specified problem with no aberrant data points. As we discussed above in Section 3.2.1, paragraph (b), the noise will still have some non-zero probability limit. We showed in Section 3.2.1, paragraph (c) that the shape will have a non-zero probability limit. And the quantity of interest $\phi(\hat{\theta})$ can generally be expected to have a non-zero probability limit. So by the decomposition of Eq. 15, if the user is interested in a question whose signal is small enough, their problem will be AMIP non-robust, despite correct specification.

(d) Though both are scaled by noise, standard errors are different from—and typically smaller than—AMIP sensitivity. Recall that classical standard errors based on limiting normal approximations also depend on $\hat{\sigma}_\psi$, in that we typically report a confidence interval for ϕ of the form $\phi \in \left(\phi(\theta, \vec{1}) \pm q_{\mathcal{N}} \frac{\hat{\sigma}_\psi}{\sqrt{N}}\right)$, where $q_{\mathcal{N}}$ is some quantile of the normal distribution, e.g. the 0.975-th quantile $q_{\mathcal{N}} \approx 1.96$. In this sense, using standard errors errors allow that ϕ may be as large as $\phi + \Delta$ whenever $\Delta/\hat{\sigma}_\psi \leq \frac{1.96}{\sqrt{N}}$. By contrast, AMIP robustness allows that ϕ may be as large as $\phi + \Delta$ when $\Delta/\hat{\sigma}_\psi \leq \hat{\mathscr{T}}_\alpha$. Since $\hat{\mathscr{T}}_\alpha \neq \frac{1.96}{\sqrt{N}}$ in general, these two approaches will yield different conclusions. Indeed, typically $\hat{\mathscr{T}}_\alpha$ converges to a non-zero constant as $N \to 0$, while $\frac{1.96}{\sqrt{N}}$ converges to zero.

(e) Statistical non-significance is always AMIP-non-robust as $N \to \infty$. This observation follows as a corollary of the discussion above. In particular, we might conclude statistical non-significance if $\left|\phi(\hat{\theta}, \vec{1})\right| \leq \frac{1.96\hat{\sigma}_\psi}{\sqrt{N}}$. To produce a statistically significant result, and so undermine the conclusion, it suffices to move $\phi(\hat{\theta}, \vec{1})$ by more than $\frac{1.96\hat{\sigma}_\psi}{\sqrt{N}}$. Take any α. As we have seen above, we can produce a change of $\hat{\sigma}_\psi \hat{\mathscr{T}}_\alpha$, which is greater than $\frac{1.96\hat{\sigma}_\psi}{\sqrt{N}}$ whenever $\hat{\mathscr{T}}_\alpha > 1.96/\sqrt{N}$. Thus, for any fixed α, there always exists a sufficiently large N such that statistical non-significance can be undermined by dropping at most α proportion of the data. By contrast, statistical significance can be robust if $\phi(\hat{\theta}, \vec{1})$ converges to a value sufficiently far from 0.

3.2.3. *The influence function*

We next review the influence function and give its particular form for Z-estimators. We establish a relationship between the influence scores and the empirical influence function. We use these connections to futher justify the relationship between the noise and the limiting distribution of $\sqrt{N}\hat{\phi}$. Finally, we use the influence function to contrast AMIP robustness with gross error robustness and establish that outliers primarily affect AMIP robustness via the noise, rather than via the shape.

(a) Writing a statistic as a functional of the empirical distribution. Before defining the influence function, we set up some useful notation. Suppose we observe IID data, $d_1, \ldots, d_N$. Each point is drawn from a data distribution $F_\infty(\cdot) = p(d_1 \leq \cdot)$, where the inequality may be multi-dimensional. For a generic distribution F, let T represent a functional of the distribution: $T(F)$. One example is the sample mean; for a generic distribution F, let $T_{mean}(F) = \int \tilde{d} \mathrm{d}F(\tilde{d})$. Then $T_{mean}(F_\infty) = \mathbb{E}[d_1]$ is the population mean. If we let $\hat{F}_N$ denote the empirical distribution function $\hat{F}_N(\cdot) = \frac{1}{N}\sum_{n=1}^N \mathbb{I}(\cdot \leq d_n)$, then $T_{mean}(\hat{F}_N) = \frac{1}{N}\sum_{n=1}^N d_n$ is the sample mean.

Now consider Z-estimators. Define $T_Z(F)$ to be a quantity satisfying

$$\int G(T_Z(F), \tilde{d}) \mathrm{d}F(\tilde{d}) = 0. \tag{19}$$

See, e.g., Hampel (1986, Section 4.2c, Def. 5). If we plug in $\hat{F}_N$ for F in Eq. 19 (and multiply both sides by N), we recover the Z-estimator estimating equation from Eq. 1, with solution $\hat{\theta} = T_Z(\hat{F}_N)$. Similarly, let $\hat{F}_w$ to be the distribution function putting weight $N^{-1}w_n$ at data point d_n. Plugging in $\hat{F}_w$ for F in Eq. 19 yields the estimating equation in Eq. 3, for weighted Z-estimators, with solution $\hat{\theta}(\vec{w}) = T_Z(\hat{F}_w)$. Finally, we can define a

new functional $T_\phi(F)$ by applying the smooth function ϕ, which picks out our quantity of interest, to $T_Z(F)$: $T_\phi(F) = \phi(T_Z(F), \vec{1})$.[10]

(b) The influence function. The influence function $\mathrm{IF}(d; T, F)$ measures the effect on a statistic T of adding an infinitesimal amount of mass at point d to some base or reference data distribution F (Reeds, 1976, Hampel, 1986). Let δ_d be the probability measure with an atom of size 1 at d. Then

$$\mathrm{IF}(d; T, F) := \lim_{\epsilon \searrow 0} \frac{T(\epsilon\delta_d + (1-\epsilon)F) - T(F)}{\epsilon}. \tag{20}$$

The influence function is defined in terms of an ordinary univariate derivative, and can be computed (as a function of d and F) using standard univariate calculus. In particular, our quantity of interest has the following influence function:

$$\mathrm{IF}(d; T_\phi, F) = -\left.\frac{\partial\phi(\theta, \vec{1})}{\partial\theta^T}\right|_{\hat{\theta}(F)} \left(\int \left.\frac{\partial G(\theta, \tilde{d})}{\partial\theta^T}\right|_{\hat{\theta}(F)} \mathrm{d}F(\tilde{d})\right)^{-1} G(\hat{\theta}(F), d). \tag{21}$$

By comparing Eq. 21 with the definition of ψ_n in Eqs. 7 and 10, we can see that, formally,[11],

$$N\psi_n = \mathrm{IF}(d_n; T_\phi, \hat{F}_N). \tag{22}$$

Eq. 22 is not a coincidence. To see this, note that the set of distributions that can be expressed as weighted empirical distributions ($\hat{F}_w$ above) is precisely the subspace of possible distribution functions concentrated on the observed data. So the derivative $N\psi_n = N\partial\phi(\hat{\theta}(\vec{w}), \vec{1})/\partial\vec{w}_n$ (Eq. 5) is simply a path derivative representation of the functional derivative $\mathrm{IF}(d_n; T_\phi, \hat{F}_N)$.

We refer to the influence function applied with $F = \hat{F}_N$ as the *empirical influence function* (Hampel, 1986). We conclude that the ψ_n that we use to form our approximation are

[10] As in ordinary calculus in Euclidean space, we can also allow for explicit F dependence in ϕ by writing $\phi(\theta, F)$. Allowing this level of generality, though, is notationally burdensome and not typical in the analysis of the influence functions for Z-estimators. So we omit this dependence for simplicity.

[11] The factor of N arises to re-write the expectation as a sum over unit-valued weights.

the values of the empirical influence function at the datapoints $d_1, \ldots, d_N$. For this reason, we refer to the ψ_n as influence scores.

(c) The sum of the influence scores is zero. We can now use standard properties of the influence function to reason about $\vec{\psi}$. For instance, the fact that $\sum_{n=1}^{N} \psi_n = 0$ follows from Eq. 21 and the fact that $\hat{\theta}$ solves Eq. 1.

(d) The noise is an estimator of the standard deviation of the limiting distribution of the quantity of interest (influence function version). Observe that, by our influence function development above, we can write the squared noise as follows.

$$\hat{\sigma}_{\psi}^2 := N \left\| \vec{\psi} \right\|_2^2 = \frac{1}{N} \sum_{n=1}^{N} (N\psi_n)^2 = \frac{1}{N} \sum_{n=1}^{N} \mathrm{IF}(d_n; T_\phi, \hat{F}_N)^2, \tag{23}$$

Recall that we saw above that $\hat{\sigma}_{\psi}^2$ consistently estimates the variance of the limiting distribution of $\sqrt{N}\hat{\phi}$, first in the special case of OLS (Section 3.1.2, paragraph (e)) and then for Z-estimators in general (Section 3.2.1, paragraph (b)). We can now see that those results are themselves special cases of the following well-known relationship between the influence function and the limiting variance of its corresponding functional:

$$\sqrt{N}\left(T(\hat{F}_N) - T(F_\infty)\right) \rightsquigarrow \mathcal{N}\left(0, \mathbb{E}\left[\mathrm{IF}(d_1; T, F_\infty)^2\right]\right), \tag{24}$$

where the expectation in the preceding display is taken with respect to $d_1 \sim F_\infty$ (see, e.g., Hampel (1986, Eq. 2.1.8)).[12] Specifically, if we can show that σ_ψ, the probability limit of $\hat{\sigma}_\psi$, is equal to $\mathbb{E}\left[\mathrm{IF}(d_1; T, F_\infty)^2\right]$, then Eq. 24 would imply $\sqrt{N}(T_\phi(\hat{F}_N) - T_\phi(F_\infty)) \rightsquigarrow \mathcal{N}(0, \sigma_\psi^2)$, just as we showed in Eq. 18 using the sandwich covariance estimator. In our case, under standard assumptions, one can show directly from Eqs. 7 and 10 that

[12] Though Eq. 24 can provide useful intuition, as it does in our case, it is often easier in any particular problem to prove asymptotic results directly rather than through the functional analysis perspective of this section, since stating precise and general conditions under which Eq. 24 holds can be challenging. See, for example, the discussion in Serfling (2009, Chapter 6) or Van der Vaart (2000, Chapter 20).

$\mathrm{IF}(d_n; T_\phi, \hat{F}_N) \xrightarrow{p} \mathrm{IF}(d_n; T_\phi, F_\infty)$, almost surely in d_n. A law of large numbers can then be applied to Eq. 23 giving the desired result.

(e) AMIP robustness is different from gross error robustness. Roughly speaking, an estimator is considered non-robust to gross errors if its influence function is unbounded (Huber, 1981). For instance, the influence function arising from the OLS Z-estimator (Section 3.1) is classically known to be non-robust to gross errors. When an influence function is unbounded, one can produce arbitrarily large changes in the quantity of interest by making arbitrarily large changes to a single datapoint. Gross-error robustness is motivated by the possibility that some small number of datapoints come from a distribution arbitrarily different from the model's posited distribution. By contrast, to assess AMIP robustness, we do not make arbitrarily large changes to datapoints. We simply remove datapoints. And the analysis is AMIP-non-robust if a change of a particular size (Δ) can be induced, rather than an arbitrarily large change. Consequently, problems with unbounded influence functions (such as OLS in Section 3.1) can be AMIP-robust if $\Delta/\hat{\sigma}_\psi$ is sufficiently large. And perfectly specified problems with no outliers can be AMIP non-robust if $\Delta/\hat{\sigma}_\psi$ is sufficiently small.

(f) Outliers affect AMIP robustness through the noise. Consideration of gross-error robustness encourages users to examine their data for unusual "outliers" in the data; once outliers are removed or their influence diminished, the problem is considered gross-error robust. Since outliers are heuristically associated with heavy-tailed data distributions, one might expect the effect of outliers to affect AMIP robustness through the shape variable $\hat{\mathscr{T}}_\alpha$. However, our analysis of Section 3.2.1, paragraph (c) shows that gross errors actually *reduce* $\hat{\mathscr{T}}_\alpha$ and so render an estimator more robust for a fixed $\hat{\sigma}_\psi$. This observation does not imply that gross errors decrease AMIP sensitivity. Rather, gross errors increase AMIP sensitivity through the noise $\hat{\sigma}_\psi$. And, as we have seen, effects on $\hat{\sigma}_\psi$ also affect the computation of standard errors.

3.3. *Accuracy of the approximation*

In Section 3.1.2, paragraph (f) we argued that our approximation was accurate in OLS for small α. Now we extend that argument to the general case. In particular, we state sufficient conditions under which $\phi^{\text{lin}}(\vec{w})$ provides a good approximation to $\phi(\hat{\theta}(\vec{w}), \vec{w})$ for small α uniformly for $\vec{w} \in W_\alpha$. Our key result, Theorem 1, holds exactly in finite samples with bounds that are, in principal, computable. Additionally, the corresponding bounds can also be expected to hold with probability approaching one as $N \to \infty$ under standard assumptions.

3.3.1. *Controlling the residual of a Taylor series*

The linear approximation we use in Eq. 5 is a Taylor series, so its accuracy can be controlled by controlling the Taylor series residual. Giordano et al. (2019b) states conditions under which the first-order Taylor series approximation to $\hat{\theta}(\vec{w})$ is accurate—precisely when using the derivative as given in Eq. 9. Under additional smoothness assumptions on ϕ, we can extend those results to our present Eq. 5. Since the Taylor series expansion is expressed in terms of observable non-asymptotic quantities, the resulting error bounds hold exactly in finite sample and are, in principle, computable.

We first state assumptions under which the linear approximation is accurate for the vector $\hat{\theta}(\vec{w})$.

ASSUMPTION 3—(Giordano et al. (2019b), Assumptions 1-4): *Let W_α be the set of weight vectors with no more than $\lfloor \alpha N \rfloor$ zeros as given by Eq. 2. Assume there exists a compact domain $\Omega_\theta \subseteq \mathbb{R}^D$ containing $\hat{\theta}(\vec{w})$ for all $\vec{w} \in W_\alpha$, such that*

1. *For all $\theta \in \Omega_\theta$ and all n, $\theta \mapsto G(\theta, d_n)$ is continuously differentiable with derivative*

$$\left.\frac{\partial G(\theta, d_n)}{\partial \theta^T}\right|_\theta =: H(\theta, d_n).$$

2. *For all $\theta \in \Omega_\theta$, there exists $C_{op} < \infty$ such that $\sup_{\theta \in \Omega_\theta} \left\| \frac{1}{N} \sum_{n=1}^N H(\theta, d_n) \right\|_{op} \leq C_{op}$.*

3. *There exists a constant $C_{gh} < \infty$ such that*

$$\sup_{\theta \in \Omega_\theta} \max \left\{ \frac{1}{N} \sum_{n=1}^{N} \|G(\theta, d_n)\|_2^2 , \frac{1}{N} \sum_{n=1}^{N} \|H(\theta, d_n)\|_2^2 \right\} \leq C_{gh}^2.$$

4. *There exists a Δ_θ and an $L_h < \infty$ such that*

$$\sup_{\theta : \left\|\theta - \hat{\theta}\right\|_2 \leq \Delta_\theta} \frac{1}{N} \sum_{n=1}^{N} \left\| H(\theta, d_n) - H(\hat{\theta}, d_n) \right\|_2^2 / \left\| \theta - \hat{\theta} \right\|_2^2 \leq L_h^2.$$

Roughly speaking, Assumption 3 states that the estimating equation is smooth and non-singular, that the sample averages are uniformly bounded, and that the estimating equation's derivatives are Lipschitz. Other than the size of the domain Ω_θ, Assumption 3 does not depend on W_α, nor on any asymptotic quantities; it states only (reasonable) assumptions on the actual problem at hand.

Under Assumption 3, we are able to apply Theorem 1 of Giordano et al. (2019b) for W_α and thereby prove the uniform accuracy of a linear approximation to $\hat{\theta}(\vec{w})$ for all $\vec{w} \in W_\alpha$. To extend the accuracy of an approximation of $\hat{\theta}(\vec{w})$ to our quantity of interest ϕ naturally requires smoothness assumptions on ϕ, which we now state.

ASSUMPTION 4: *Define the re-scaled weights $\delta_n := \vec{w}_n / \sqrt{N}$, and assume that $\theta, \delta \mapsto \phi(\theta, \sqrt{N}\delta)$ has continuous partial derivatives, that the partial derivatives' $\|\cdot\|_2$-norm evaluated at $\theta = \hat{\theta}(\vec{1})$ and $\vec{w} = \vec{1}$ is bounded by a finite constant C_ϕ, and that the partial derivatives are Lipschitz in $\|\cdot\|_2$ with finite constant L_ϕ.*

We can now state our main accuracy theorem.

THEOREM 1: *Let Assumptions 3 and 4 hold. For sufficiently small α, there exist constants C_1 and C_2, defined in terms of quantities given in Assumptions 3 and 4, such that*[13]

$$\sup_{\vec{w} \in W_\alpha} \left\| \phi^{\text{lin}}(\vec{w}) - \phi(\hat{\theta}(\vec{w}), \vec{w}) \right\| \leq C_1 \alpha \quad \textit{and} \quad \sup_{\vec{w} \in W_\alpha} \left\| \phi(\hat{\theta}(\vec{w}), \vec{w}) - \hat{\phi} \right\| \leq C_2 \sqrt{\alpha}. \quad (25)$$

[13] We note that the rate $\sqrt{\alpha}$ is determined by a simple but coarse Cauchy-Schwartz bound (see Lemma 2). Tighter bounds may be achievable when the random variables $\|G(\theta, d_n)\|_2$ and $\|H(\theta, d_n)\|_2$ are uniformly integrable (see, e.g., Section 2.5 of Van der Vaart (2000)).

When α is small, we expect $\alpha \ll \sqrt{\alpha}$ (for example, when $\alpha = 0.01$, $\sqrt{\alpha} = 0.1 \gg 0.01$), so Theorem 1 states that the bound in the error of our linear approximation shrinks faster than the bound in the function itself as $\alpha \to 0$.

Theorem 1 is a finite-sample result, applying exactly to the problem at hand. All else equal, finite-sample results are preferable to asymptotic ones. Nevertheless, due to the many loose bounds employed in the proof, we do not expect the constants to be useful in practice. Additionally, Theorem 1 of Giordano et al. (2019b) may in theory require α to be smaller than $1/N$, resulting in a vacuous statement. Improving these shortcomings is an important avenue for future work (e.g. Giordano et al. (2019a), Wilson et al. (2020)). But it is therefore useful to observe that, when uniform laws of large numbers apply to $\theta \mapsto \|G(\theta, \cdot)\|_2$ and $\theta \mapsto \|H(\theta, \cdot)\|_2$, and the limiting functions are also non-singular, bounded, and Lipschitz, then one can expect Assumption 3 to hold with high probability and finite constants as $N \to \infty$. A precise statement of the necessary conditions for such asymptotics to apply is given in Lemma 1 of Giordano et al. (2019b).

3.3.2. *Limitations of linear approximations*

In every case we examine in our applications in Section 4, we manually re-run the analysis without the data points in the removal set $\hat{S}_\alpha$; in doing so, we find that the change suggested by the approximation is nearly always achieved in practice (a notable exception is given and discussed at the end of Section 4.4). However, linear approximations are only approximations, and intuition about the potential weaknesses of linear approximations in general apply to our approximation. The crux of Theorem 1 is that small α implies that $\vec{w} - \vec{1}$ is small, thus we can control the error of a linear approximation in $\vec{w}$ evaluated at $\vec{1}$. Conversely, one would not expect the approximation to work well in general for large α and the correspondingly larger $\vec{w} - \vec{1}$.

As an extreme example, consider when the linear approximation reports that there is no feasible way to effect a particular change; i.e., when $\hat{\alpha}^*_\Delta = \texttt{NA}$ (see Definition 2). Such a result may seem to imply that, no matter how many datapoints one removes, the estimator will not change by an amount Δ, which is often absurd. However, such a result should be taken to mean that one would have to remove such a large proportion α of datapoints

that the linear approximation on which we are basing the $\hat{\alpha}^*_\Delta$ is invalid. A more accurate interpretation of $\hat{\alpha}^*_\Delta = \texttt{NA}$ is that no *small* proportion of points can be removed to produce a change Δ, for if there were such a small proportion, the linear approximation would have discovered it.

Similarly, linear approximations cannot be expected to work well near the boundary of parameter spaces. For example, if the quantity of interest is a variance, then the true parameter is constrained to be positive, but our linear approximation is not. It can help to linearize the problem using unconstrained reparameterizations (e.g., linearly approximating the log variance rather than variance). However, as we show in Section 4.4, simply transforming to an unconstrained space is still not guaranteed to produce accurate approximations near the boundary in the original, constrained space.

4. APPLIED EXPERIMENTS

4.1. *The Oregon Medicaid experiment*

In our first experiment, we show that even empirical analyses that display little classical uncertainty can be sensitive to the removal of less than 1% of the sample. We consider the Oregon Medicaid study (Finkelstein et al., 2012) and focus on health outcomes. The standard errors of the treatment effects are small relative to effect size; against a null hypothesis of no effect, most p values are well below 0.01. Yet we find that for most of the results, removing less than 1% of the sample can produce a significant result of the opposite sign to the full-sample analysis. In one case, removing less than 0.05% of the sample can change the significance of the result.

4.1.1. *Background and replication*

First we provide some context for the analysis and results of Finkelstein et al. (2012). In early 2008, the state of Oregon opened a waiting list for new enrollments in its Medicaid program for low-income adults. Oregon officials then drew names by lottery from the 90,000 people who signed up, and those who won the lottery could sign up for Medicaid along with any of their household members. This setup created a randomization into treatment and control groups at the household level. The Finkelstein et al. (2012) study

measures outcomes one year after the treatment group received Medicaid. About 25% of the treatment group did indeed have Medicaid coverage by the end of the trial. The main analysis investigates treatment assignment as treatment itself ("intent to treat" or ITT analysis) and uses treatment assignment as an instrumental variable for take-up of insurance coverage ("local average treatment effect" or LATE analysis).

We focus on the health outcomes of winning the Medicaid lottery, which appear in Panel B from Table 9 of Finkelstein et al. (2012). Each of these J outcomes is denoted by y_{ihj} for individual i in household h for outcome type j. The data sample to which we have access consists of survey responders ($N = 23{,}741$); some responders are from the same household. The variable $\texttt{LOTTERY}_h$ equals one if household h won the Medicaid lottery, and zero otherwise. All regressions use a set of covariates X_{ih} comprised of household size fixed effects, survey wave fixed effects, and the interaction between the two. All regressions also use a set of demographic and economic covariates V_{ih}. To infer the ITT effects of winning the Medicaid lottery, the authors estimate the following model via OLS:

$$y_{ihj} = \beta_0 + \beta_1 \texttt{LOTTERY}_h + \beta_2 X_{ih} + \beta_3 V_{ih} + \epsilon_{ihj}.$$

To infer the LATE of taking up Medicaid on compliers, the authors employ an Instrumental Variables (IV) strategy using the lottery as an instrument for having Medicaid insurance. All standard errors are clustered on the household, and all regressions are weighted using survey weights defined by the variable `weight_12m`. We have access to the following seven outcome variables, presented in Panel B of Table 9 of the original paper (as well as our tables below) in the following order: a binary indicator of a self-reported measure of health being good or very good or excellent (not fair or poor), a binary indicator of self-reported health not being poor, a binary indicator of health being about the same or improving over the last six months, the number of days of good physical health in the past 30 days, the number of days on which poor physical or mental health did not impair usual activities, the number of days mental health was good in the past 30 days, and an indicator of not being depressed in last two weeks. We replicate Panel B of Table 9 of Finkelstein et al. (2012) exactly, both for the ITT effect ($\hat{\beta}_1$) for the entire population and for the LATE

Study case	Original estimate	Target change	Refit estimate	Observations dropped
Health genflip 12m	0.133 (0.026)*	Sign change	**-0.006 (0.025)**	275 = 1.18%
		Significance change	**0.044 (0.026)**	162 = 0.69%
		Significant sign change	-0.043 (0.024)	381 = 1.63%
Health notpoor 12m	0.099 (0.018)*	Sign change	**-0.003 (0.015)**	155 = 0.66%
		Significance change	**0.027 (0.016)**	100 = 0.43%
		Significant sign change	**-0.030 (0.015)***	219 = 0.94%
Health change flip 12m	0.113 (0.023)*	Sign change	**-0.006 (0.022)**	197 = 0.84%
		Significance change	**0.039 (0.022)**	106 = 0.45%
		Significant sign change	**-0.049 (0.022)***	291 = 1.24%
Not bad days total 12m	1.317 (0.563)*	Sign change	**-0.023 (0.535)**	73 = 0.33%
		Significance change	**1.078 (0.558)**	10 = 0.05%
		Significant sign change	-1.009 (0.521)	144 = 0.66%
Not bad days physical 12m	1.585 (0.606)*	Sign change	**-0.040 (0.577)**	87 = 0.41%
		Significance change	**1.131 (0.597)**	20 = 0.09%
		Significant sign change	**-1.141 (0.566)***	164 = 0.77%
Not bad days mental 12m	2.082 (0.640)*	Sign change	**-0.062 (0.607)**	123 = 0.57%
		Significance change	**1.171 (0.625)**	42 = 0.19%
		Significant sign change	**-1.201 (0.594)***	212 = 0.98%
Nodep Screen 12m	0.078 (0.025)*	Sign change	**-0.005 (0.024)**	123 = 0.53%
		Significance change	**0.046 (0.024)**	42 = 0.18%
		Significant sign change	**-0.050 (0.023)***	220 = 0.95%

TABLE I

MEDICAID PROFIT RESULTS WITH IV FOR A RANGE OF OUTCOME VARIABLES. THE "REFIT ESTIMATE" COLUMN SHOWS THE RESULT OF RE-FITTING THE MODEL REMOVING THE APPROXIMATE MOST INFLUENTIAL SET. STARS INDICATE SIGNIFICANCE AT THE 5% LEVEL. REFITS THAT ACHIEVED THE DESIRED CHANGE ARE BOLDED.

on compliers ($\hat{\pi}_1$). Both analyses show strong evidence for positive effects on all health measures, with most p values well below 0.01.

4.1.2. *AMIP Sensitivity Results*

For each health outcome in Panel B from Table 9 of Finkelstein et al. (2012), we compute the AMIP to assess how many data points one needs to remove to change the sign of the treatment effect, the significance of the treatment effect, or produce a significant result

Study case	Original estimate	Target change	Refit estimate	Observations dropped
Health genflip 12m	0.039 (0.008)*	Sign change	**-0.004 (0.008)**	286 = 1.22%
		Significance change	**0.013 (0.008)**	163 = 0.70%
		Significant sign change	**-0.021 (0.008)***	422 = 1.81%
Health notpoor 12m	0.029 (0.005)*	Sign change	**-0.001 (0.005)**	156 = 0.67%
		Significance change	**0.008 (0.005)**	101 = 0.43%
		Significant sign change	**-0.009 (0.004)***	224 = 0.96%
Health change flip 12m	0.033 (0.007)*	Sign change	**-0.002 (0.006)**	198 = 0.85%
		Significance change	**0.011 (0.007)**	106 = 0.45%
		Significant sign change	**-0.015 (0.006)***	292 = 1.25%
Not bad days total 12m	0.381 (0.162)*	Sign change	**-0.013 (0.157)**	74 = 0.34%
		Significance change	**0.306 (0.161)**	11 = 0.05%
		Significant sign change	**-0.309 (0.153)***	147 = 0.67%
Not bad days physical 12m	0.459 (0.175)*	Sign change	**-0.017 (0.169)**	88 = 0.41%
		Significance change	**0.328 (0.172)**	20 = 0.09%
		Significant sign change	**-0.344 (0.165)***	166 = 0.78%
Not bad days mental 12m	0.603 (0.184)*	Sign change	**-0.027 (0.178)**	124 = 0.57%
		Significance change	**0.340 (0.181)**	42 = 0.19%
		Significant sign change	**-0.381 (0.175)***	216 = 1.00%
Nodep Screen 12m	0.023 (0.007)*	Sign change	**-0.001 (0.007)**	123 = 0.53%
		Significance change	**0.013 (0.007)**	43 = 0.19%
		Significant sign change	**-0.015 (0.007)***	225 = 0.97%

TABLE II

MEDICAID PROFIT RESULTS WITH OLS FOR A RANGE OF OUTCOME VARIABLES. THE "REFIT ESTIMATE" COLUMN SHOWS THE RESULT OF RE-FITTING THE MODEL REMOVING THE APPROXIMATE MOST INFLUENTIAL SET. STARS INDICATE SIGNIFICANCE AT THE 5% LEVEL. REFITS THAT ACHIEVED THE DESIRED CHANGE ARE BOLDED.

of the opposite sign. The sensitivity of the LATE analysis is shown in Table I and the sensitivity of the ITT analysis is shown in Table II. In both cases we use exactly the models from the original paper, with all fixed effects and controls included and with clustering at the household level. For most outcomes, for both the LATE and ITT analysis, the sign of the treatment effect can be changed by removing around 0.5% of the data, or approximately 100 data points in a sample of approximately 22,000. The most robust outcome, "Health being better than fair" ("Health genflip 12m"), requires the removal of a little over 1%

of the sample to change the sign. Across the various outcomes, we can drop even less of the sample to change the results from significant to non-significant. In some cases, we need remove only 10 or 20 data points to effect a change in significance. Finally, for most outcomes, we can remove less than 1% of the data to produce a significant result of the opposite sign. The only two exceptions, "Health genflip 12m" and "Health change flip 12m", require the removal of slightly more than 1% to generate a significant result with the opposite sign.

We check the performance of the approximation for each analysis by re-running the model after manually removing the data points in the Approximate Most Influential Set. The result of this procedure is shown in the "Refit Estimate" column of Tables I and II. For almost every result in each table, our approximate metric reliably uncovers combinations of data points that do deliver the claimed changes. As we discuss in Section 2.2.1, the changes recorded in the "Refit Estimate" column of Tables I and II form a lower bound on the true worst-case finite-sample sensitivity.

By comparing Table I with Table II, we see that the ITT results, estimated via OLS, are not notably more AMIP-robust than the LATE results, which are estimated via IV. This may seem at first counterintuitive based on a heuristic belief that IV is in some sense a less "robust" analysis than OLS in finite sample: for example, recent authors, including Young (2019), have suggested that the uncertainty intervals for IV may be more poorly calibrated in finite samples than the intervals for OLS. However, as we discuss in Section 3, the quality of being "robust" in the sense of a finite-sample estimator providing a good approximation to an asymptotic quantity is simply unrelated to AMIP robustness. Neither the size of the AMIP itself nor the accuracy of the AMIP approximation depends on asymptotic arguments (see, e.g., Section 3.2.2, paragraph (a) and the discussion of Theorem 1). The AMIP measures the sensitivity to data ablation of a particular procedure on a particular dataset and is indifferent to the fidelity of the chosen quantity of interest to some asymptotic limit. For this reason, a procedure such as IV may be "non-robust" in the sense of having poor coverage in finite sample (as reported by Young (2019)) and yet be AMIP-robust, or vice versa—the two notions of "robustness" are simply different.

4.2. *Cash transfers*

We next show that an empirical analysis can still be AMIP-non-robust even after outliers are removed. To that end, we apply our techniques to examine the robustness of the main analysis from Angelucci and De Giorgi (2009), one of the flagship studies showing the impact of cash transfers on ineligible ("non-poor") households in the same villages, also known as "spillover effects." The authors trimmed the consumption outcome for the non-poor households due to concerns about the influence of the largest values. Yet while the analysis on the poor households is quite robust, the analysis on the non-poor households—whom the trimming protocol actually affects—is much more sensitive.

4.2.1. *Background and replication*

Angelucci and De Giorgi (2009) employ a randomized controlled trial to study the impact of Progresa, a social program giving cash gifts to eligible poor households in Mexico. The randomization occurs at the village level. So one can estimate both a main effect on the poor households selected to receive Progresa and also the impact on the non-eligible "non-poor" households located in the same villages as Progresa-receiving poor households.

The main results of the paper show that there are strong positive impacts of Progresa on total household consumption measured as an index both for eligible poor households and for the non-eligible households; see Table 1 of Angelucci and De Giorgi (2009). The variable $\texttt{C\_ind}_{it}$ denotes total household consumption for household i in time period t. Values of $\texttt{C\_ind}_{it}$ above 10,000 are removed; such households are, by definition, non-poor. The authors study three different time periods separately to detect any change in the impact between the short and long term. They condition on a large set of variables (a household poverty index, land size, head of household gender, age, whether the household speaks an indigenous language, and literacy; at the locality level, a poverty index, and the number of households) to help ensure a fair comparison between households in the treatment and control villages. In this case these controls are important; the effects on the "non-poor" households are significant at the 5% level when the controls are included, but

they are only significant at the 10% level in a simple regression on a dummy for treatment status.

The full data for the paper is available on the website of the *American Economic Review* thanks to the open-data policies of the journal and the authors. We can successfully replicate the results of this analysis with the controls and without, and we proceed with the controls in our present analysis in accordance with the original authors' preferred specification. We consider the time periods indexed as $t = 8, 9, 10$ in the dataset provided, though we note that the authors do not rely on the results at $t = 8$ as the roll-out was still ongoing. We employ K control variables, where X_{itk} is the k-th variable for household i in period t. Then we run the following regression:

$$\texttt{C\_ind}_{it} = \beta_0 + \beta_1 \texttt{treat}_{poor,i} + \beta_2 \texttt{treat}_{nonpoor,i} + \sum_{k=1}^{K} \beta_{2+k} X_{itk} + \epsilon_{it}.$$

Here, $\texttt{treat}_{poor,i}$ refers to an interaction between the treatment indicator and an indicator for being a poor household; correspondingly, $\texttt{treat}_{nonpoor,i}$ is an interaction between the treatment indicator and an indicator for being a non-poor household. We are able to exactly replicate the results of Table 1 of Angelucci and De Giorgi (2009), which exhibits positive effects of cash transfers.

4.2.2. *AMIP Sensitivity Results*

We apply our methodology to assess how many data points one need remove to change the sign, the significance, or to generate a significant result of the opposite sign to that found in the full sample. We focus on the latter two time periods, as households had received only partial transfers in the first time period, but we show all three in order to replicate Table 1 from the original paper. Table III shows our results. Focusing on periods 9 and 10, we find that the inferences on the direct effects on the poor households are quite robust, but the inferences on the indirect effects are less so. For the analysis of the poor, one typically needs to remove much more than 1% of the sample to change conclusions. For the analysis of the non-poor, we can remove less than 0.5% of the data to change conclusions. In fact, we

Study case	Original estimate	Target change	Refit estimate	Observations dropped
Poor, period 8	17.312 (4.576)*	Sign change	**-0.656 (3.745)**	252 = 2.30%
		Significance change	**7.284 (4.087)**	83 = 0.76%
		Significant sign change	**-7.212 (3.443)***	464 = 4.24%
Poor, period 9	27.924 (5.770)*	Sign change	**-1.377 (4.406)**	345 = 3.58%
		Significance change	**7.077 (4.555)**	146 = 1.52%
		Significant sign change	**-8.951 (4.251)***	588 = 6.11%
Poor, period 10	33.861 (4.468)*	Sign change	**-2.559 (3.541)**	697 = 6.63%
		Significance change	**4.806 (3.684)**	435 = 4.14%
		Significant sign change	**-9.416 (3.296)***	986 = 9.37%
Non-poor, period 8	-5.444 (7.133)	Sign change	**0.260 (6.410)**	5 = 0.11%
		Significance change	-12.845 (6.635)	16 = 0.35%
		Significant sign change	9.670 (5.573)	24 = 0.52%
Non-poor, period 9	22.852 (10.000)*	Sign change	**-0.365 (7.542)**	21 = 0.55%
		Significance change	**16.506 (9.114)**	3 = 0.08%
		Significant sign change	-11.733 (7.113)	53 = 1.38%
Non-poor, period 10	21.493 (9.405)*	Sign change	**-0.573 (6.750)**	30 = 0.70%
		Significance change	**16.262 (8.927)**	3 = 0.07%
		Significant sign change	-10.845 (6.467)	92 = 2.16%

TABLE III

CASH TRANSFERS RESULTS FOR VARIOUS PERIODS AND TREATMENT GROUPS. THE "REFIT ESTIMATE" COLUMN SHOWS THE RESULT OF RE-FITTING THE MODEL REMOVING THE APPROXIMATE MOST INFLUENTIAL SET. STARS INDICATE SIGNIFICANCE AT THE 5% LEVEL. REFITS THAT ACHIEVED THE DESIRED CHANGE ARE BOLDED.

can remove only 3 data points in a sample of approximately 10,000 households to change the significance status for both $t = 9$ and $t = 10$.

We again check the quality of our approximation. The "Refit Estimate" column in Table III shows the results of manually re-running each analysis after removing the implicated data points. In most cases the AMIP correctly identifies a combination of data points that can make the claimed changes to the conclusions of the study. Although there are a few cases where re-running the analysis fails to produce the predicted statistically significant sign change, the observed changes are still large enough to be of practical interest. Fur-

thermore, it is likely that the removal of a few additional points would in fact produce the desired statistically significant sign reversals.

Finally, we note that these results constitute an illustration of how gross error robustness is distinct from AMIP robustness (see Section 3.2.3, paragraph (e)). Recall that Angelucci and De Giorgi (2009) removed (non-poor) datapoints for which consumption was greater than 10,000. By removing outliers of the consumption variable in this way, the authors of this study made what is typically considered a conservative choice in view of classical robustness concerns about gross error sensitivity. Yet, as we have shown in Table III, qualitative conclusions concerning the non-poor households remain non-robust to the removal of a small number of datapoints, which demonstrates empirically that one cannot necessarily make an analysis AMIP-robust by simply trimming outliers. Indeed, as we showed above in Section 3.1, even perfectly specified OLS regressions with no aberrant data points can be AMIP-non-robust if the signal to noise ratio is too low.

4.3. *Seven RCTs of microcredit: Linear regression analysis*

We now show that even a simple 2-parameter linear model that performs a comparison of means between the treatment and control group of a randomized trial can be highly sensitive. To that end, we consider the analysis of seven randomized controlled trials of expanding access to microcredit, first aggregated in Meager (2019). In Section 4.4 below, we will consider a more complicated Bayesian hierarchical model on the same data.

4.3.1. *Background*

Each of the seven microcredit studies was conducted in a different country, and each study selected certain communities to randomly receive greater access to microcredit. Researchers either built a branch, or combined building a branch with some active outreach, or randomly selected borrowers among those who applied. The selected studies are: Angelucci et al. (2015), Attanasio et al. (2015), Augsburg et al. (2015), Banerjee et al. (2015), Crépon et al. (2015), Karlan and Zinman (2011), and Tarozzi et al. (2015). Six of these studies were published in a special issue of the *American Economics Journal: Applied Economics*

on microcredit. All seven studies together are commonly considered to represent the most solid evidence base for understanding the impact of microcredit.

We follow the original studies and Meager (2019) in analyzing the impact of access to microcredit as the treatment of interest. The studies range in their sample sizes from around 1,000 households in Mongolia (Attanasio et al., 2015) to around 16,500 households in Mexico (Angelucci et al., 2015). We first focus on the headline results on household business profit regressed on an intercept and a binary variable indicating whether a household was allocated to the treatment group or to the control group. For household i in site k, let Y_{ik} denote the profit measured, and let T_{ik} denote the treatment status. We estimate the following model via OLS:

$$Y_{ik} = \beta_0 + \beta T_{ik} + \epsilon_{ik}. \tag{26}$$

This regression model compares the means in the treatment and control groups and estimates the difference as $\hat{\beta}$. We follow Meager (2019) in omitting the control variables or fixed effects from the regressions in order to examine the robustness of this fundamental procedure. But in principle this omission should make no difference to the estimate $\hat{\beta}$, and indeed it does not (Meager, 2019).[14]

4.3.2. *AMIP sensitivity results*

The sensitivity results for the linear regression of profit on microcredit access appear in Table IV. In all cases, by removing less than 1% of the data points can change either the sign or the significance. In three of the studies, one can drop less than 1% of the data points to generate a result of the opposite sign that would be deemed significant at the 5% level. Mexico, the largest study, is the most sensitive: a single data point among the 16,561

[14] The omission may in principle make a difference to the inference on β by affecting the standard errors. However, it turns out that in these studies the additional covariates make very little difference to the standard errors. We also do not cluster the standard errors at the community level for the same reason; the results are not substantially changed. Running the regression above in each of the seven studies delivers almost identical results to the preferred specification, as it should if intra-cluster correlations are weak and covariates are not strongly predictive of household profit.

Study case	Original estimate	Target change	Refit estimate	Observations dropped
Bosnia	37.534 (19.780)	Sign change	**-2.226 (15.628)**	14 = 1.17%
		Significance change	**43.732 (18.889)***	1 = 0.08%
		Significant sign change	**-34.929 (14.323)***	40 = 3.35%
Ethiopia	7.289 (7.893)	Sign change	**-0.053 (2.513)**	1 = 0.03%
		Significance change	**15.356 (7.763)***	45 = 1.45%
		Significant sign change	**-8.755 (1.852)***	66 = 2.12%
India	16.722 (11.830)	Sign change	**-0.501 (8.221)**	6 = 0.09%
		Significance change	**22.895 (10.267)***	1 = 0.01%
		Significant sign change	**-16.638 (7.537)***	32 = 0.47%
Mexico	-4.549 (5.879)	Sign change	**0.398 (3.194)**	1 = 0.01%
		Significance change	**-10.962 (5.565)***	14 = 0.08%
		Significant sign change	**7.030 (2.549)***	15 = 0.09%
Mongolia	-0.341 (0.223)	Sign change	**0.021 (0.184)**	16 = 1.66%
		Significance change	**-0.436 (0.220)***	2 = 0.21%
		Significant sign change	**0.361 (0.147)***	38 = 3.95%
Morocco	17.544 (11.401)	Sign change	**-0.569 (9.920)**	11 = 0.20%
		Significance change	**21.720 (11.003)***	2 = 0.04%
		Significant sign change	**-18.847 (9.007)***	30 = 0.55%
Philippines	66.564 (78.127)	Sign change	**-4.014 (57.204)**	9 = 0.81%
		Significance change	**138.929 (66.880)***	4 = 0.36%
		Significant sign change	**-122.494 (49.409)***	58 = 5.21%

TABLE IV

MICROCREDIT REGRESSIONS FOR THE PROFIT OUTCOME. THE "REFIT ESTIMATE" COLUMN SHOWS THE RESULT OF RE-FITTING THE MODEL REMOVING THE APPROXIMATE MOST INFLUENTIAL SET. STARS INDICATE SIGNIFICANCE AT THE 5% LEVEL. REFITS THAT ACHIEVED THE DESIRED CHANGE ARE BOLDED.

households in Mexico determines the sign (as also discussed above in Section 2.4). To produce a statistically significant result of the opposite sign—that is, to turn Mexico's noisy negative result into a "strong" positive result—one need remove only 15 data points, less than 0.1% of the sample. Mongolia, the smallest study in terms of sample size, is among the most robust in terms of sign changes; it takes 2% of the sample to change the sign. Producing a significant result of the opposite sign also requires more than 1% removal in the Philippines, Bosnia, Ethiopia, and Mongolia—whereas Mexico, India, and Morocco are more sensitive. We check the performance of our approximation by manually re-running

Study case	Original estimate	Target change	Refit estimate	Observations dropped
Bosnia	-5.803 (2.819)*	Sign change	**0.395 (2.135)**	10 = 1.00%
		Significance change	**-4.870 (2.693)**	1 = 0.10%
		Significant sign change	**5.130 (1.978)***	33 = 3.31%
India	-1.643 (0.576)*	Sign change	**0.035 (0.506)**	41 = 0.60%
		Significance change	-1.051 (0.536)*	8 = 0.12%
		Significant sign change	**1.059 (0.487)***	85 = 1.25%
Mexico	-0.082 (0.094)	Sign change	**0.000 (0.091)**	12 = 0.07%
		Significance change	**-0.180 (0.091)***	14 = 0.09%
		Significant sign change	**0.176 (0.087)***	55 = 0.33%
Mongolia	1.523 (2.103)	Sign change	**-0.033 (0.973)**	3 = 0.31%
		Significance change	**2.717 (1.027)***	10 = 1.04%
		Significant sign change	**-2.623 (0.689)***	45 = 4.68%
Morocco	-0.420 (0.723)	Sign change	**0.047 (0.669)**	3 = 0.05%
		Significance change	**-1.351 (0.667)***	14 = 0.26%
		Significant sign change	**1.252 (0.602)***	23 = 0.42%

TABLE V

MICROCREDIT REGRESSIONS FOR THE TEMPTATION OUTCOME. THE "REFIT ESTIMATE" COLUMN SHOWS THE RESULT OF RE-FITTING THE MODEL REMOVING THE APPROXIMATE MOST INFLUENTIAL SET. STARS INDICATE SIGNIFICANCE AT THE 5% LEVEL. REFITS THAT ACHIEVED THE DESIRED CHANGE ARE BOLDED.

the analysis with the data removed; the "Refit Estimate" column shows that the claimed reversal is always achieved in practice for these analyses.

By comparing the results of the present section with those of Sections 4.1 and 4.2, we can confirm the conclusion of Section 3.2.2, paragraph (d) that standard errors are, in general, distinct from AMIP sensitivity. Despite the fact that original estimates of Table IV are statistically insignificant, some of these non-significant results are more AMIP-robust than some of the significant results in the Cash Transfers and Oregon Medicaid examples; consider the "Significant sign change" result in the Philippines study, for example.

We further demonstrate that the AMIP sensitivity observed in Table IV cannot simply be ascribed to statistical insignificance by considering a different outcome with smaller variability, showing that it reveals a similar sensitivity to the profit outcome. The variable we now consider is household consumption spending on temptation goods such as alcohol,

chocolate, and cigarettes, since the effect of microcredit on temptation spending was estimated by Meager (2019) with the greatest precision of all six considered outcome variables. Table V shows the results of applying the AMIP to the same regression given in Eq. 26, but with temptation spending as the outcome. While somewhat more robust than the profit analyses, the difference in the approximate removal proportions in Table V is not large.

Finally, one might be tempted to ascribe the AMIP-non-robust results in Table IV to outliers resulting from the heavy tails of the household profit variable (a phenomenon well-documented by Meager (2020)). However, as we discuss in Section 3.2.3, paragraph (e) above, gross error robustness is qualitatively distinct from AMIP robustness (see also the discussion of outlier trimming at the end of Section 4.2). Indeed, the more complex hierarchical model of the next section, Section 4.4, was designed precisely to accommodate the heavy tail of the household profit variable, and yet—as we will show—still exhibits a high degree of AMIP-sensitivity.

4.4. *Seven RCTs of microcredit: Bayesian hierarchical tailored mixture model*

In this section, we investigate a Bayesian hierarchical model, both demonstrating that even Bayesian analyses can exhibit considerable AMIP sensitivity, and showing an example of a parameter of interest for which our linear approximation performs badly. We specifically focus on a variational Bayes approximation to the tailored mixture model from Meager (2020). One might hope that any of the following aspects of the more complicated model might alleviate AMIP sensitivity: the use of hierarchical Bayesian evidence aggregation, the regularization from incorporation of priors, or the somewhat more realistic data-generating process captured in this specific tailored likelihood. Indeed, the approach of Meager (2020) was specifically motivated by the desire to capture important features of the data-generating process such as heavier tails. On the contrary, we find that the average estimated effects of microcredit remain sensitive according to the AMIP, as we did in the simpler models of Section 4.3. We also find that the linear approximation that underlies the AMIP performs poorly when attempting to decrease a particular hypervariance param-

eter, providing a concrete example of the limitations of our methodology, particularly for parameters near the boundary of the set of their allowable values.

4.4.1. *Background*

Following Meager (2020), we fit a hierarchical model (hereafter referred to as the "microcredit model") to all the data from the seven microcredit RCTs. We model each outcome using a spike at zero and two lognormal tail distributions, one for the positive realizations of profit and one for the negative realizations. Within the model, microcredit can affect the proportion of data assigned to each of these three components as well as affecting the location and scale of the lognormal tails. There is a hierarchical shrinkage element to the model for each parameter. The hypervariances of the treatment effects are of particular interest because these capture heterogeneity in effects across studies and offer information about the transportability of results across settings.

The models in the original paper were fit via Hamiltonian Monte Carlo (HMC) with the software package Stan (Carpenter et al., 2017). It is possible to compute the Approximate Maximum Influence Perturbation for HMC, or for any Markov Chain Monte Carlo method, using the tools of Bayesian local robustness (Gustafson, 2000, Giordano et al., 2018), but the sensitivity of simulation-based estimators is beyond the scope of this paper. However, there are ways to estimate Bayesian posteriors via Z-estimators, such as with Variational Bayes (VB) techniques (Blei et al., 2017).[15] Specifically, we fit the microcredit model using a variant of Automatic Differentiation Variational Inference (ADVI) described in Giordano et al. (2018, Section 5.2) (see also the original ADVI paper, Kucukelbir et al. (2017)). Since the posterior uncertainty estimates of vanilla ADVI are notoriously inaccurate, we estimated posterior uncertainty using linear response covariances, again following Giordano et al. (2018, Section 5.2).[16] We verified that the posterior means and covariance estimates produced by our variational procedure and the corresponding estimates from run-

[15] The Laplace approximation can also be expressed as a Z-estimator.

[16] When forming the Approximate Most Influential Set, we approximated the sensitivity only of the posterior means to data removal; the linear response covariances were considered fixed. However, when we report the results of re-fitting the model, we did re-calculate the linear response covariances at the new variational optimum.

Model parameter	Original estimate	Target change	Refit estimate	Observations dropped
τ_-	0.102 (0.070)	Sign change	**-0.042 (0.090)**	31 = 0.09%
		Significance change	0.138 (0.071)	11 = 0.03%
		Significant sign change	-0.204 (0.106)	99 = 0.28%
τ_+	0.078 (0.033)*	Sign change	**-0.021 (0.046)**	74 = 0.21%
		Significance change	**0.062 (0.033)**	9 = 0.03%
		Significant sign change	-0.100 (0.054)	163 = 0.46%

TABLE VI

MICROCREDIT MIXTURE RESULTS FOR A SELECTED SET OF MODEL PARAMETERS. STANDARD ERRORS AND "SIGNIFICANCE" ARE BASED ON THE ESTIMATED 95% POSTERIOR CREDIBLE INTERVALS. THE "REFIT ESTIMATE" COLUMN SHOWS THE RESULT OF RE-FITTING THE MODEL REMOVING THE APPROXIMATE MOST INFLUENTIAL SET. STARS INDICATE SIGNIFICANCE AT THE 5% LEVEL. REFITS THAT ACHIEVED THE DESIRED CHANGE ARE BOLDED.

Model parameter	Original estimate	Change type	Refit estimate	Prediction	Observations dropped
$\log \sigma_{\tau_-}$	-2.313	Drop 0.5% to increase	0.126	-0.811	177 = 0.50%
		Drop 0.5% to decrease	-0.151	-4.066	177 = 0.50%
$\log \sigma_{\tau_+}$	-3.100	Drop 0.5% to increase	-1.095	-1.204	177 = 0.50%
		Drop 0.5% to decrease	-1.598	-4.974	177 = 0.50%

TABLE VII

RESULTS FOR THE LOG POSTERIOR STANDARD DEVIATION ESTIMATES OF THE EFFECT SIZE DISTRIBUTION IN THE MICROCREDIT MIXTURE MODEL. SIGN AND SIGNIFICANCE ARE NOT MEANINGFUL FOR POSTERIOR STANDARD DEVIATIONS, SO WE DROP 0.5% OF DATAPOINTS TO ATTEMPT TO PRODUCE LARGE POSITIVE AND NEGATIVE CHANGES. THE "REFIT ESTIMATE" COLUMN SHOWS THE RESULT OF RE-FITTING THE MODEL REMOVING APPROXIMATE MOST INFLUENTIAL SET. THE "PREDICTION" COLUMN SHOWS THE PREDICTED CHANGE UNDER THE SAME PERTURBATION.

ning HMC with Stan were within reasonable agreement relative to the posterior standard deviation.

4.4.2. *AMIP Sensitivity Results*

We first consider the effect of microcredit on the location parameter of the positive and negative tails of profit, given respectively by the parameters τ_+ and τ_-. Roughly speaking, τ_+ and τ_- are both estimating the effect of microcredit averaged across all of the seven

countries analyzed in Section 4.3. Our point estimates for τ_+ and τ_- are given by their respective VB posterior means. We used the linear response covariance estimates to form a 95% posterior credible interval in place of confidence intervals, and consider a change "significant" if the posterior credible interval does not contain zero.

Table VI shows the sensitivity of inference concerning τ_+ and τ_-. We see that the microcredit model's estimates of the average effectiveness of microcredit remain highly sensitive to the removal of small percentages of the sample, despite being derived from a model that accounts for non-Gaussian data shape and is regularized by the priors. This sensitivity shows that Bayesian aggregation procedures do not necessarily produce AMIP-robust estimates.

We next examine the sensitivity of the hypervariances, which measure the variability of the effect of microcredit on these tails from country to country. Specifically, the parameters $\sigma^2_{\tau_+}$ and $\sigma^2_{\tau_-}$ represent the between-country variances of the effect of microcredit on positive and negative profit outcomes, respectively. The σ parameter can be thought of as the scale parameter analogue of the corresponding location parameter τ from Table VI. The hypervariances are of particular practical interest because they quantify how variable the effect of microcredit might be—small values of the hypervariance imply that all countries respond similarly to microcredit, whereas large values imply that one should not necessarily extrapolate the efficacy of microcredit from one country to another.

In order to avoid the possibility of extrapolating to negative variances, we form a linear approximation to our variational Bayes estimates of the posterior mean of $\log \sigma$. Since $\log \sigma$ is a scale parameter measuring the variability from country to country of the effect of microcredit, its sign is not particularly meaningful, nor is it particularly interesting to ask whether its posterior credible interval contains zero. Rather, we are interested in the magnitude of $\log \sigma$. So, to investigate robustness, we use the AMIP to check the approximate maximum change achievable in either direction (increasing or decreasing the magnitude of $\log \sigma$) by removing 0.5% of the sample, about the same fraction of the data as could generate a "significant" sign change for the $\tau_\pm$ parameters.

The results for the hypervariances, given in Table VII, represent a useful demonstration of the limitations of our linear approximation. We are able to find sets of datapoints which, when dropped, produce *increases* in the hypervariances, though the our linear approximation is not nearly as accurate as in the rest of our results above. When we attempted to drop points in order to decrease the hypervariances, however, the linear approximation failed utterly—the Approximate Most Influential Set designed to produce a decrease in the hypervariances instead produced a large *increase* upon refitting.

Given that the hypervariances are constrained to be positive, our failure to produce large decreases may not be surprising. Note that the hypervariances' posterior expectations began very small, and that decreasing them pushes the posterior of the hypervariances closer to the boundary of the admissible space. Though the log variance may in principle take arbitrarily negative values, it nevertheless appears that the model exhibits strongly non-linear dependence on the data weights for very small variances. Designing useful diagnostics for detecting and explaining such deviations from nonlinearity in complex models is an interesting avenue for future work. In the meantime, Table VII shows the importance, when possible, of checking the accuracy of the AMIP predictions by refitting the model, and of exercising caution when using the AMIP approximation near the boundary of the parameter space.

5. CONCLUSION

There are different ways of quantifying the dependence between the finite-sample realization and the conclusions of statistical inference. While this dependence has become synonymous with standard errors in frequentist statistics, the notions are equivalent only under a certain paradigm that considers a hypothetical perfect random resampling exercise for the purpose of evaluating a specific parameter within a given model. This hypothetical may not capture all the data sensitivity relevant to applied social science. Much of 20th century statistics, with its focus on standard errors and sampling uncertainty, has its origins in the context of randomized agricultural trials, where the difference in yield across multiple fields is well-modeled by independent sampling variation. Contrast with trials of economic interventions to alleviate poverty, where randomly sampling individuals or communities

is a challenge and interventions may be applied across very different contexts. In applied economics, statistical models are often intended to provide tractable and interpretable summaries of the impact of interventions. In doing so, models can average information across individuals in ways that may not reflect policy interests.[17] For our methods to safely inform economic policy decisions, then, we need additional tools beyond standard errors.

In this paper, we have offered one alternative way of conceiving of and quantifying the dependence of empirical results on the sample data, beyond standard errors. Sensitivity of conclusions to data removal under our metric does not necessarily imply a problem with the sample. But the goal of inference is not to learn about the sample, but rather to learn about the population. If minor alterations to the sample can generate major changes in the inference, and we know that the environment in which we do economics is changing all the time, we ought to be less confident that we have learned something fundamental about this broader population we seek to understand, for whom we ultimately seek to make policy. We do not mean to imply that the original analysis is invalid according to classical sampling theory, and we do not recommend that researchers abandon the original full-sample results even if they are not robust according to our metric. However, reporting our metrics alongside standard errors would improve our ability to understand and interpret the findings of a given analysis.

Since AMIP analysis always indicates which data points have high (approximate) influence, our methods allow researchers not only the chance to check that the approximation worked on their own sample, but to understand what—if anything—makes these data points special. Investigating influential points may provide insight into the way in which a given inferential procedure is using the finite-sample information to generate claims about the population parameters. In addition, in cases when this sensitivity is undesirable, it may be fruitful to develop new statistical methods to ameliorate it. It seems particularly important

[17] In agricultural trials, total yields are the true quantity of interest; for microcredit trials, the average treatment effect is but a convenient summary. If the average profit were to increase slightly through one individual becoming wealthy while leaving all others destitute, one could consider the intervention a failure. By contrast, if a single plant produced an entire harvest's worth of corn, the outcome would still be desirable, if strange.

to develop these methods in view of the actual goals and uses of economics research, rather than relying on a classical resampling paradigm that bears little resemblance to the practice of applied social science.

REFERENCES

Angelucci, M. and De Giorgi, G. (2009). Indirect effects of an aid program: How do cash transfers affect ineligibles' consumption? *American Economic Review*, 99(1):486–508.

Angelucci, M., Karlan, D., and Zinman, J. (2015). Microcredit impacts: Evidence from a randomized microcredit program placement experiment by Compartamos Banco. *American Economic Journal: Applied Economics*, 7(1):151–82.

Attanasio, O., Augsburg, B., De Haas, R., Fitzsimons, E., and Harmgart, H. (2015). The impacts of microfinance: Evidence from joint-liability lending in Mongolia. *American Economic Journal: Applied Economics*, 7(1):90–122.

Augsburg, B., De Haas, R., Harmgart, H., and Meghir, C. (2015). The impacts of microcredit: Evidence from Bosnia and Herzegovina. *American Economic Journal: Applied Economics*, 7(1):183–203.

Banerjee, A., Duflo, E., Glennerster, R., and Kinnan, C. (2015). The miracle of microfinance? Evidence from a randomized evaluation. *American Economic Journal: Applied Economics*, 7(1):22–53.

Baydin, A., Pearlmutter, B., Radul, A., and Siskind, J. (2017). Automatic differentiation in machine learning: A survey. *The Journal of Machine Learning Research*, 18(1):5595–5637.

Blei, D., Kucukelbir, A., and McAuliffe, J. (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877.

Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., and Riddell, A. (2017). Stan: A probabilistic programming language. *Journal of Statistical Software*, 76(1).

Chatterjee, S. and Hadi, A. (1986). Influential observations, high leverage points, and outliers in linear regression. *Statistical Science*, 1(3):379–393.

Chen, X., Tamer, E., and Torgovitsky, A. (2011). Sensitivity analysis in semiparametric likelihood models. *Cowles Foundation discussion paper*.

Crépon, B., Devoto, F., Duflo, E., and Parienté, W. (2015). Estimating the impact of microcredit on those who take it up: Evidence from a randomized experiment in Morocco. *American Economic Journal: Applied Economics*, 7(1):123–50.

De Bruijn, N. (1981). *Asymptotic methods in analysis*, volume 4. Courier Corporation.

Finkelstein, A., Taubman, S., Wright, B., Bernstein, M., Gruber, J., Newhouse, J., Allen, H., Baicker, K., and Group, O. H. S. (2012). The Oregon health insurance experiment: Evidence from the first year. *The Quarterly Journal of Economics*, 127(3):1057–1106.

Giordano, R., Broderick, T., and Jordan, M. I. (2018). Covariances, robustness and variational Bayes. *The Journal of Machine Learning Research*, 19(1):1981–2029.

Giordano, R., Jordan, M. I., and Broderick, T. (2019a). A higher-order Swiss army infinitesimal jackknife. *arXiv preprint arXiv:1907.12116*.

Giordano, R., Stephenson, W., Liu, R., Jordan, M. I., and Broderick, T. (2019b). A Swiss army infinitesimal jackknife. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1139–1147. PMLR.

Gustafson, P. (2000). Local robustness in Bayesian analysis. In *Robust Bayesian Analysis*, pages 71–88. Springer.

Hampel, F. (1974). The influence curve and its role in robust estimation. *Journal of the American Statistical Association*, 69(346):383–393.

Hampel, F. (1986). *Robust statistics: the approach based on influence functions*, volume 196. Wiley-Interscience.

Hansen, L. and Sargent, T. (2008). *Robustness*. Princeton University Press.

He, X., Jurečková, J., Koenker, R., and Portnoy, S. (1990). Tail behavior of regression estimators and their breakdown points. *Econometrica: Journal of the Econometric Society*, pages 1195–1214.

Huber, P. (1967). The behavior of maximum likelihood estimates under nonstandard conditions. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability: Weather modification*, volume 5, page 221. Univ of California Press.

Huber, P. (1981). *Robust Statistics*. John Wiley & Sons, New York.

Karlan, D. and Zinman, J. (2011). Microcredit in theory and practice: Using randomized credit scoring for impact evaluation. *Science*, 332(6035):1278–1284.

Krantz, S. and Parks, H. (2012). *The implicit function theorem: History, theory, and applications*. Springer Science & Business Media.

Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., and Blei, D. (2017). Automatic differentiation variational inference. *The Journal of Machine Learning Research*, 18(1):430–474.

Leamer, E. (1984). Global sensitivity results for generalized least squares estimates. *Journal of the American Statistical Association*, 79(388):867–870.

Leamer, E. (1985). Sensitivity analyses would help. *The American Economic Review*, 75(3):308–313.

Maclaurin, D., Duvenaud, D., and Adams, R. (2015). Autograd: Effortless gradients in numpy. In *ICML 2015 AutoML Workshop*, volume 238.

Masten, M. and Poirier, A. (2020). Inference on breakdown frontiers. *Quantitative Economics*, 11(1):41–111.

Meager, R. (2019). Understanding the average impact of microcredit expansions: A Bayesian hierarchical analysis of seven randomized experiments. *American Economic Journal: Applied Economics*, 11(1):57–91.

Meager, R. (2020). Aggregating distributional treatment effects: A Bayesian hierarchical analysis of the microcredit literature. *LSE working paper*.

Mosteller, F. and Tukey, J. (1977). *Data Analysis and Regression: A Second Course In Statistics*. Pearson, USA.

Reeds, J. (1976). *On the definition of von Mises functionals*. PhD thesis, Statistics, Harvard University.

Rigollet, P. (Spring 2015). Course 18.s997 High Dimensional Statistics. Massachusetts Institute of Technology: MIT OpenCouseWare, https://ocw.mit.edu/. Accessed: 2021-08-24.

Saltelli, A. (2004). Global sensitivity analysis: An introduction. In *Proc. 4th International Conference on Sensitivity Analysis of Model Output (SAMO '04)*, pages 27–43.

Serfling, R. (2009). *Approximation theorems of mathematical statistics*, volume 162. John Wiley & Sons.

Sobol, I. (2001). Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation*, 55(1-3):271–280.

Stefanski, L. and Boos, D. (2002). The calculus of M-estimation. *The American Statistician*, 56(1):29–38.

Tarozzi, A., Desai, J., and Johnson, K. (2015). The impacts of microcredit: Evidence from Ethiopia. *American Economic Journal: Applied Economics*, 7(1):54–89.

Van der Vaart, A. (2000). *Asymptotic statistics*, volume 3. Cambridge University Press.

Wilson, A., Kasy, M., and Mackey, L. (2020). Approximate cross-validation: Guarantees for model assessment and selection. In *International Conference on Artificial Intelligence and Statistics*, pages 4530–4540. PMLR.

Young, A. (2019). Consistency without inference: Instrumental variables in practical application. http://personal.lse.ac.uk/YoungA/CWOI.pdf. Accessed: 2020-11-27.

Yu, B. (2013). Stability. *Bernoulli*, 19(4):1484–1500.

APPENDIX A DETAILED PROOFS

LEMMA 1: *Let* $\chi_1, \ldots, \chi_N$ *be real-valued scalars with* $\frac{1}{N}\sum_{n=1}^N \chi_n^2 = 1$. *Then* $\max_{\vec{w}\in W_\alpha} \frac{1}{N}\sum_{n=1}^N |\vec{w}_n - 1|\chi_n \leq \sqrt{\alpha}$.

PROOF: Without loss of generality, let the χ_n be unique (if they are not, add an arbitrarily small amount of jitter to break ties), and let $q_{1-\alpha}$ denote their $\lceil (1-\alpha)N \rceil$-th largest value. The maximum $\max_{\vec{w}\in W_\alpha} \frac{1}{N}\sum_{n=1}^N |\vec{w}_n - 1|\chi_n$ is achieved at $\vec{w}$ which sets to zero the weights of all $\{n : \chi_n \geq q_{1-\alpha}\}$, so

$$\max_{\vec{w}\in W_\alpha} \frac{1}{N}\sum_{n=1}^N |\vec{w}_n - 1|\chi_n = \frac{1}{N}\sum_{n=1}^N \mathbb{I}\left(\chi_n \geq q_{1-\alpha}\right)\chi_n.$$

Let $\hat{F}_\chi$ denote the empirical distribution on χ_n conditional on the data d_n, and note that $q_{1-\alpha}$ is fixed in $\hat{F}_\chi$. Applying Cauchy-Schwartz to the preceding display with the distribution $\hat{F}_\chi$ gives

$$\frac{1}{N}\sum_{n=1}^N \mathbb{I}\left(\chi_n \geq q_{1-\alpha}\right)\chi_n \leq \sqrt{\frac{1}{N}\sum_{n=1}^N \mathbb{I}\left(\chi_n \geq q_{1-\alpha}\right)^2}\sqrt{\frac{1}{N}\sum_{n=1}^N \chi_n^2} = \sqrt{\frac{\lfloor N\alpha \rfloor}{N}} \leq \sqrt{\alpha},$$

since $\frac{1}{N}\sum_{n=1}^N \chi_n^2 = 1$ and at most $\lfloor \alpha N \rfloor$ points are greater than $q_{1-\alpha}$. *Q.E.D.*

The following lemma satisfies Condition 1 of Giordano et al. (2019b).

LEMMA 2: *Let* $W_\alpha^* := \{\vec{1} + t(\vec{w} - \vec{1}) : \vec{w} \in W_\alpha, t \in [0,1]\}$ *Under Assumption 3,*

$$\max_{\vec{w}\in W_\alpha^*} \sup_{\theta\in\Omega_\theta} \left\| \frac{1}{N}\sum_{n=1}^N (\vec{w}_n - 1)G(\theta, d_n) \right\|_1 \leq \sqrt{D}C_{gh}\sqrt{\alpha} \quad \textit{and}$$

$$\max_{\vec{w}\in W_\alpha^*} \sup_{\theta\in\Omega_\theta} \left\| \frac{1}{N}\sum_{n=1}^N (\vec{w}_n - 1)H(\theta, d_n) \right\|_1 \leq \sqrt{D}C_{gh}\sqrt{\alpha}.$$

PROOF: We prove the result for $G(\theta, d_n)$; the proof for $H(\theta, d_n)$ follows analogously. By the triangle inequality and the relationship between $\|\cdot\|_2$ and $\|\cdot\|_1$,

$$\left\| \frac{1}{N}\sum_{n=1}^N (\vec{w}_n - 1)G(\theta, d_n) \right\|_1 \leq \sqrt{D}C_{gh}\frac{1}{N}\sum_{n=1}^N |\vec{w}_n - 1|\frac{\|G(\theta, d_n)\|_2}{C_{gh}}.$$

Apply Lemma 1 with $\chi_n := \frac{\|G(\theta,d_n)\|_2}{C_{gh}}$ to control the maximum of the sum over W_α. Finally, the results extends to W_α^* since

$$\max_{\vec{w}\in W_\alpha^*} \frac{1}{N}\sum_{n=1}^N |\vec{w}_n - 1|\chi_n = \max_{t\in[0,1]} \max_{\vec{w}\in W_\alpha} \frac{1}{N}\sum_{n=1}^N |t(\vec{w}_n - 1)|\chi_n = \max_{\vec{w}\in W_\alpha} \frac{1}{N}\sum_{n=1}^N |(\vec{w}_n - 1)|\chi_n.$$

Q.E.D.

We need the following lemma to extend the result of Giordano et al. (2019b), Theorem 1 to smooth functions.

LEMMA 3: *Let Assumptions 3 and 4 hold. For sufficiently small α, there exists a constant $C_b < \infty$ such that, for any $a \in \mathbb{R}^N$,*

$$\max_{\vec{w}\in W_\alpha^*} \left\| \left(\left.\frac{d\hat{\theta}(\vec{w})}{d\vec{w}^T}\right|_{\vec{w}} - \left.\frac{d\hat{\theta}(\vec{w})}{d\vec{w}^T}\right|_{\vec{1}} \right) a \right\|_2 \leq C_b \frac{\|a\|_2}{\sqrt{N}}\sqrt{\alpha}.$$

PROOF: As in the proof of Theorem 1, for the remainder of the proof assume that $\alpha \leq \frac{\Delta^2}{DC_{gh}^2}$, and observe that Assumptions 1-5 and Condition 1 of Giordano et al. (2019b) are satisfied. For the duration of this proof, define the shorthand notation

$$H(\vec{w}) := \frac{1}{N}\sum_{n=1}^N \vec{w}_n H(\hat{\theta}(\vec{w}), d_n) \quad \text{and} \quad G(\vec{w}) := \frac{1}{N}\sum_{n=1}^N a_n G(\hat{\theta}(\vec{w}), d_n).$$

Then, by the indicated results from Giordano et al. (2019b),

$$\begin{aligned}
&\left\| \left(\left.\frac{d\hat{\theta}(\vec{w})}{d\vec{w}^T}\right|_{\vec{w}} - \left.\frac{d\hat{\theta}(\vec{w})}{d\vec{w}^T}\right|_{\vec{1}} \right) a \right\|_2 \\
&\quad = \left\| -H(\vec{w})^{-1}G(\vec{w}) + H(\vec{1})^{-1}G(\vec{1}) \right\|_2 \quad \text{(Proposition 4)} \\
&\quad \leq \left\| -(H(\vec{w})^{-1} - H(\vec{1})^{-1})G(\vec{w}) \right\|_2 + \left\| H(\vec{1})^{-1}(G(\vec{1}) - G(\vec{w})) \right\|_2 \\
&\quad \leq \left\| -(H(\vec{w})^{-1} - H(\vec{1})^{-1})G(\vec{w}) \right\|_2 + C_{op}\delta. \quad \text{(Condition 1, Assumption 2)}
\end{aligned}$$

Then,

$$\left\| (H(\vec{w})^{-1} - H(\vec{1})^{-1})G(\vec{w}) \right\|_2$$

$$
\begin{aligned}
&= \left\| H(\vec{w})^{-1} \left(H(\vec{1}) - H(\vec{w}) \right) H(\vec{1})^{-1} G(\vec{w}) \right\|_2 \\
&\leq 2C_{op}^2 \left\| \left(H(\vec{1}) - H(\vec{w}) \right) G(\vec{w}) \right\|_2 \quad \text{(Assumption 2, Lemma 6)} \\
&\leq 2C_{op}^2 \sqrt{D}(1 + DC_w L_h C_{op}) \delta \left\| G(\vec{w}) \right\|_2 \quad \text{(Lemma 5, Matrix norms)} \\
&= 2C_{op}^2 \sqrt{D}(1 + DC_w L_h C_{op}) \delta \left\| \frac{1}{N} \sum_{n=1}^{N} a_n G(\hat{\theta}(\vec{w}), d_n) \right\|_2 \\
&\leq 2C_{op}^2 \sqrt{D}(1 + DC_w L_h C_{op}) \delta C_{gh} \frac{\|a\|_2}{\sqrt{N}}. \quad \text{(Assumption 3, Cauchy-Schwartz)}
\end{aligned}
$$

Combining, and using our Lemma 2 to give $\delta = \sqrt{D} C_{gh} \sqrt{\alpha}$, gives the desired result. *Q.E.D.*

Proof of Theorem 1. For the duration of the proof, define the linear approximation $\hat{\theta}^{\text{lin}}(\vec{w}) := \hat{\theta} + \frac{\mathrm{d}\hat{\theta}(\vec{w})}{\mathrm{d}\vec{w}^T}\Big|_{\vec{1}} (\vec{w} - \vec{1})$. Assumption 3 is equivalent to Assumptions 1-4 of Giordano et al. (2019b), and Lemma 2 satisfies Condition 1 of Giordano et al. (2019b) with $\delta = \sqrt{D} C_{gh} \sqrt{\alpha}$. Assumption 5 of Giordano et al. (2019b) is satisfied for W_α with $C_w = 1$. Define, as in Giordano et al. (2019b), $C_{IJ} := 1 + DL_h C_{op}$ and $\Delta := \min\left\{ \Delta_\theta C_{op}^{-1}, \frac{1}{2} C_{op}^{-1} C_{IJ}^{-1} \right\}$, So Lemma 3 and Theorem 1 of Giordano et al. (2019b) give, respectively, that

$$
\max_{\vec{w} \in W_\alpha^*} \left\| \hat{\theta}(\vec{w}) - \hat{\theta} \right\|_2 \leq C_{op} \sqrt{D} C_{gh} \sqrt{\alpha} \quad \text{and} \tag{27}
$$

$$
\alpha \leq \frac{\Delta^2}{D C_{gh}^2} \quad \Rightarrow \quad \max_{\vec{w} \in W_\alpha^*} \left\| \hat{\theta}^{\text{lin}}(\vec{w}) - \hat{\theta}(\vec{w}) \right\|_2 \leq 2C_{op}^2 C_{IJ} D C_{gh}^2 \alpha. \tag{28}
$$

For the remainder of the proof assume that $\alpha \leq \frac{\Delta^2}{D C_{gh}^2}$ so that Eq. 28 applies.

For any $\vec{w} \in W_\alpha$, define $\omega(t) := \vec{1} + t(\vec{w} - \vec{1}) \in W_\alpha^*$. By the fundamental theorem of calculus,

$$
\phi(\hat{\theta}(\vec{w}), \vec{w}) - \hat{\phi} = \int_0^1 \frac{\mathrm{d}\phi(\omega(t))}{\mathrm{d}t}\Big|_t dt = \int_0^1 \left(\frac{\mathrm{d}\phi(\omega(t))}{\mathrm{d}t}\Big|_t - \frac{\mathrm{d}\phi(\omega(t))}{\mathrm{d}t}\Big|_1 \right) \mathrm{d}t + \frac{\mathrm{d}\phi(\omega(t))}{\mathrm{d}t}\Big|_1 . \tag{29}
$$

where, by the chain rule,

$$\left.\frac{\mathrm{d}\phi(\omega(t)))}{\mathrm{d}t}\right|_t = \left.\frac{\partial\phi(\theta,\omega(t))}{\partial\theta^T}\right|_{\hat{\theta}(\omega(t))} \left.\frac{\mathrm{d}\hat{\theta}(\vec{w})}{\mathrm{d}\vec{w}^T}\right|_{\omega(t)} (\vec{w}-\vec{1}) + \left.\frac{\partial\phi(\hat{\theta}(\omega(t)),\vec{w})}{\partial\vec{w}^T}\right|_{\omega(t)} (\vec{w}-\vec{1}).$$

It will be useful to adopt a specific "big O" notation for the remainder of the proof, by which we mean the following. If we write $x = O(\sqrt{\alpha})$ for some quantity x, we mean that there exists a constant C, available as a closed-form function of constants defined in Assumptions 3 and 4, such that $x \leq C\sqrt{\alpha}$ for all $\alpha \leq \frac{\Delta^2}{DC_{gh}^2}$. An analogous notation meaning is given to $x = O(\alpha)$. This "big O" notation can be manipulated in the usual ways (De Bruijn, 1981).

To begin with, by definition of W_α, we have $\max_{\vec{w}\in W_\alpha} \frac{1}{N}\sum_{n=1}^N (\vec{w}_n - 1)^2 = \frac{\lfloor \alpha N \rfloor}{N} \leq \alpha$, so $\max_{\vec{w}\in W_\alpha} \left\|(\vec{w}-\vec{1})/\sqrt{N}\right\|_2 \leq \sqrt{\alpha}$.

Next, observe that Eqs. 27 and 28 together imply that

$$\max_{\vec{w}\in W_\alpha^*} \left\| \left.\frac{\mathrm{d}\hat{\theta}(\vec{w})}{\mathrm{d}\vec{w}^T}\right|_{\vec{1}} (\vec{w}-\vec{1}) \right\|_2 \leq \max_{\vec{w}\in W_\alpha^*} \left\|\hat{\theta}^{\mathrm{lin}}(\vec{w}) - \hat{\theta}(\vec{w})\right\|_2 + \max_{\vec{w}\in W_\alpha^*} \left\|\hat{\theta}(\vec{w}) - \hat{\theta}\right\|_2 = O(\sqrt{\alpha}).$$

By Lemma 3 below, we have that

$$\max_{t\in[0,1]} \max_{\vec{w}\in W_\alpha^*} \left\| \left(\left.\frac{\mathrm{d}\hat{\theta}(\vec{w})}{\mathrm{d}\vec{w}^T}\right|_{\omega(t)} - \left.\frac{\mathrm{d}\hat{\theta}(\vec{w})}{\mathrm{d}\vec{w}^T}\right|_{\vec{1}} \right) (\vec{w}-\vec{1}) \right\|_2 \leq C_b \frac{\left\|\vec{w}-\vec{1}\right\|_2}{\sqrt{N}} \sqrt{\alpha} = O(\alpha).$$

Combining the previous two displays gives, by the triangle inequality, that $\max_{t\in[0,1]} \max_{\vec{w}\in W_\alpha^*} \left\| \left.\frac{\mathrm{d}\hat{\theta}(\vec{w})}{\mathrm{d}\vec{w}^T}\right|_{\omega(t)} (\vec{w}-\vec{1}) \right\|_2 = O(\sqrt{\alpha})$.

Finally, by the Lipschitz property of the partial derivatives in Assumption 4, we have that

$$\max_{t\in[0,1]} \left\| \left.\frac{\partial\phi(\theta,\omega(t))}{\partial\theta}\right|_{\hat{\theta}(\omega(t))} - \left.\frac{\partial\phi(\theta,\vec{1})}{\partial\theta}\right|_{\hat{\theta}} \right\|_2 = O(\sqrt{\alpha}) \quad \text{and}$$

$$\max_{t\in[0,1]} \sqrt{N} \left\| \left.\frac{\partial\phi(\hat{\theta}(\omega(t)),\vec{w})}{\partial\vec{w}}\right|_{\omega(t)} - \left.\frac{\partial\phi(\hat{\theta},\vec{w})}{\partial\vec{w}}\right|_{\vec{1}} \right\|_2 = O(\sqrt{\alpha}).$$

Again, the triangle inequality with the boundedness of the partial derivatives of ϕ at $\vec{w} = \vec{1}$ implies

$$\max_{t\in[0,1]} \left\| \left. \frac{\partial\phi(\theta,\omega(t))}{\partial\theta} \right|_{\hat{\theta}(\omega(t))} \right\|_2 \quad \text{and} \quad \max_{t\in[0,1]} \sqrt{N} \left\| \left. \frac{\partial\phi(\hat{\theta}(\omega(t)),\vec{w})}{\partial\vec{w}} \right|_{\omega(t)} \right\|_2 = O(\sqrt{\alpha}).$$

Combining the above results gives that

$$\max_{t\in[0,1]} \left\| \left. \frac{d\phi(\omega(t))}{dt} \right|_t \right\|_2 = O(\sqrt{\alpha}) \quad \text{and} \quad \max_{t\in[0,1]} \left\| \left. \frac{\mathrm{d}\phi(\omega(t))}{\mathrm{d}t} \right|_t - \left. \frac{\mathrm{d}\phi(\omega(t))}{\mathrm{d}t} \right|_1 \right\|_2 = O(\alpha),$$

from which the desired conclusion follows by Eq. 29. *Q.E.D.*

Evaluating Sensitivity to the Stick-Breaking Prior in Bayesian Nonparametrics

Ryan Giordano [*†] Runjing Liu [*‡]
Michael I. Jordan [‡] Tamara Broderick [†]

October 26, 2021

Bayesian models based on the Dirichlet process and other stick-breaking priors have been proposed as core ingredients for clustering, topic modeling, and other unsupervised learning tasks. Prior specification is, however, relatively difficult for such models, given that their flexibility implies that the consequences of prior choices are often relatively opaque. Moreover, these choices can have a substantial effect on posterior inferences. Thus, considerations of robustness need to go hand in hand with nonparametric modeling. In the current paper, we tackle this challenge by exploiting the fact that variational Bayesian methods, in addition to having computational advantages in fitting complex nonparametric models, also yield sensitivities with respect to parametric and nonparametric aspects of Bayesian models. In particular, we demonstrate how to assess the sensitivity of conclusions to the choice of concentration parameter and stick-breaking distribution for inferences under Dirichlet process mixtures and related mixture models. We provide both theoretical and empirical support for our variational approach to Bayesian sensitivity analysis.

1 Introduction

Scientists and engineers working in a wide range of fields are often interested in inferring the number of clusters in a given data set, as well as inferring which data points belong together. Such inferential questions can be posed naturally within a Bayesian nonparametric (BNP) framework, building on tools such as the Dirichlet process [Ferguson, 1973, Sethuraman, 1994]. The Dirichlet process has two useful attributes that have made it be suggested as a natural model of clustering phenomena. First, it is a combinatorial stochastic process, exhibiting discrete structure that allows multiple data points to be associated with the same

[*]Equal contribution author
[†]Department of EECS, MIT
[‡]Department of Statistics, UC Berkeley

underlying value of a parameter. Second, its nonparametric nature means that the number of unique parameter values generally grows with the size of the data set, accommodating growth in the number of inferred clusters as data accrue. Such growth is appropriate in many real-world settings; for example, we might expect to keep discovering new species as we examine more individual organisms, and we might expect to discover more topics as we read more articles in a scientific literature. Finally, the overall Bayesian framework in which the Dirichlet process is embedded allows clustering to be treated as one aspect of a larger inferential problem. In particular, the Dirichlet process can be flexibly incorporated into more complex models that exhibit other forms of structure, including hierarchical, spatio-temporal, and topological structure.

Although the BNP framework offers flexibility, it is important to recognize that it is not a black-box method. As with any Bayesian methodology, the deployment of a BNP model involves choices of hyperparameters. Often, these choices are made for reasons of mathematical or computational convenience. Indeed, the nonparametric nature of BNP models can make it particularly difficult to express prior belief subjectively. For example, the latent frequencies of clusters provided by the Dirichlet process are obtained by recursively removing beta-distributed fractions of probability mass from the unit interval. The use of the beta distribution is motivated by its mathematical tractability under recursion and by the fact that it yields a form of conditional conjugacy that can be exploited by Gibbs sampling. These are appealing properties, but it is difficult to imagine justifying this specific choice subjectively, particularly given that observable consequences of the choice are indirect. Even having accepted the beta distribution as a choice of convenience, there remains the problem of choosing the parameter α associated with this distribution. The implications of this choice are again difficult to assess subjectively. In practice the choice is often made based on previous applications or by simply employing a heuristic [Teh et al., 2006, Gelman et al., 2013, Chapter 23].

In summary, it is important to recognize that there will exist many possible values of α, and many possible forms of stick-breaking prior, that might correspond to one's prior beliefs, but which the Dirichlet process framework and other complex BNP models bundle in a way that makes it difficult to understand and to specify *a priori*. Choices of convenience are therefore made, and, unfortunately, these choices can change the results of a data analysis. For instance, α has a direct, proportional relationship to the number of clusters obtained asymptotically in draws from the Dirichlet process. Thus the number of clusters inferred at any particular data size may depend strongly on α. If our scientific conclusions varied substantially because of such dependence, we might worry that these conclusions were driven not by the data and meaningful prior beliefs but instead by our arbitrary or default choices. It behooves us, then, to check how sensitive our conclusions are to these choices.

The outputs of Bayesian inference arise not just from a model and collection of data but also via the use of some posterior approximation. Accordingly, when we assess sensitivity, we should assess the sensitivity of this full procedure to our

model choices. In the current paper we focus on Dirichlet process mixture (DPM) models and Variational Bayesian (VB) posterior approximations based on reverse Kulback-Leibler (KL) divergence. VB methods have several favorable properties that motivate their use in the DPM setting. First, they exhibit fast computational scaling due to their use of gradient-based optimization. Second, they avoid the label-switching problem exhibited by MCMC in the mixture-model setting [Jasra et al., 2005]. Third, their implementation has become increasingly straightforward due to automatic differentiation tools [Ranganath et al., 2013, Kucukelbir et al., 2016]. Finally, and of particular interest in the current paper, the variational formulation makes it possible to compute closed-form derivative-based expansions of posterior distributions as a function of model hyperparameters [Giordano et al., 2018]. Thus VB provides a natural pathway to quantifying the robustness of Bayesian inference.

Concretely, with a fully specified model and inference procedure in hand in the setting of DPM models, we can ask how sensitive some quantity of interest is to the choices of α and the stick-breaking distributions. One option is to propose a number of potential α values, compute the variational approximation at each α value, and report our quantity of interest for each α value. We might similarly assess sensitivity to the stick-breaking distribution over a range of distributional choices. There are at least two major issues with this proposal: (1) while VB is a relatively fast form of approximate Bayesian inference in general, it may still be prohibitively expensive to have to re-run it many times, and (2) it is unclear how best to choose a collection of α and (especially) the stick-breaking distribution values—and how many to choose.

In this work, we address these challenges by making full use of the variational nature of VB methodology. We show how to approximate the nonlinear dependence of the VB optimum on prior choices using a first-order Taylor series expansion. We build on the local robustness tools developed by Giordano et al. [2018] for VB and Gustafson [1996a] for the exact posterior and MCMC approximations. To enable their application to DPM models, we solve a number of open problems: (1) we establish that the optimal VB parameters are a continuously differentiable function of α and a particular parametrization of the stick-breaking form; (2) we show that the sensitivity of the VB approximation to functional prior perturbations takes the form of an integral against a computationally tractable *influence function*—and illustrate how the influence function can provide an interpretable summary of the effect of arbitrary changes to the prior density; (3) to justify using linear approximations over a ball describing different stick-breaking densities, we show that our method is a *uniformly* good approximation by establishing Fréchet differentiability; (4) we show how to compute our approximation efficiently in high-dimensional problems; and (5) we establish the accuracy, practicality, and computational efficiency of our approximation for a variety of models that use stick-breaking, and for various quantities of interest in both clustering and topic modeling. Though our present focus is on DPM models, many of our results apply to VB approximations in general.

The remainder of the paper is organized as follows. In Section 2, we review the stick-breaking construction of the Dirichlet process and our chosen variational approximation. In Section 3, we derive the form of local prior robustness measures for VB approximations. We consider functional perturbations to the stick-breaking density in Section 4, and define the influence function from which we can construct influential and worst-case perturbations. We review related work in Section 5. In Section 6, we address scalability and other computational considerations for computing local sensitivity on real applications. In Section 7, we apply our tools to assess the sensitivity of BNP models in several data analysis problems.

2 The Model and Variational Approximation

2.1 A stick-breaking model for clustering

Consider a standard Bayesian nonparametric generative model for clustering, with observed data $x = (x_n)_{n=1}^N$. We assume a countable infinity of latent components, with frequencies $\pi = (\pi_1, \pi_2, \ldots)$, such that $\pi_k \in [0,1]$ for all $k \in \{1, 2, \ldots\}$, and $\sum_k \pi_k = 1$. For the nth data point, the vector $z_n = (z_{n1}, z_{n2}, \ldots)$ is an indicator vector; $z_{nk} = 1$ represents the assignment of the nth data point to the kth component, with all other vector elements set equal to zero. We generate $z_{nk} = 1$ with probability π_k, i.i.d. across n. To generate the x_n, we assume the kth component is characterized by a component-specific parameter, $\beta_k \in \Omega_\beta \subseteq \mathbb{R}^{D_\beta}$, and that a data point arising from component k is generated as $\mathcal{P}(x_n|\beta_k)$. Then $\mathcal{P}(x_n|z_n, \beta) = \prod_{k=1}^\infty \mathcal{P}(x_n|\beta_k)^{z_{nk}}$. The β_k in turn are generated i.i.d. from a prior $\mathcal{P}_{\text{base}}(\beta_k)$. For instance, in a Gaussian mixture model, β_k could be a vector representing the mean and covariance of a Gaussian distribution.

It remains to place a prior on the component frequencies π. We will focus on stick-breaking priors for π, so we first replace π with a stick-breaking representation. Let $\nu = (\nu_1, \nu_2, \ldots)$ represent proportions: $\nu_k \in [0,1]$. Take

$$\pi_k := \nu_k \prod_{k'<k} (1 - \nu_{k'}). \tag{1}$$

We then define a stick-breaking prior by placing a prior on the ν_k. Fix a density, $\mathcal{P}_{\text{stick}}(\cdot)$, with respect to the Lebesgue measure on $[0,1]$ and let $\nu_k \overset{iid}{\sim} \mathcal{P}_{\text{stick}}(\nu_k)$ for $k \in \{1, 2, \ldots\}$. A common choice of $\mathcal{P}_{\text{stick}}$ is Beta$(1, \alpha)$, with *concentration parameter* $\alpha > 0$. With this choice, the π are distributed according to the size-biased weights associated with the atoms of a draw from a Dirichlet process. This particular beta stick-breaking prior is often favored due to its convenient mathematical properties and ease of use in inference.

Posterior quantities of interest. In theory, with our generative model and observed data in hand, we can find the Bayesian posterior $\mathcal{P}(\beta, z, \nu|x)$ and report any posterior summaries of interest. For instance, the posterior $\mathcal{P}(\beta, z, \nu|x)$ induces a posterior distribution on the number of clusters $G_{\text{cl}}(z)$, where *clusters* are

components to which at least one data point has been assigned:

$$G_{\text{cl}}(z) := \sum_{k=1}^{\infty} \mathbb{I}\left(\left(\sum_{n=1}^{N} z_{nk}\right) > 0\right),$$

where $\mathbb{I}(\cdot)$ is the indicator function taking value 1 when the argument is true and 0 otherwise.

In practice, though, neither the posterior nor the posterior summary is readily accessed. An approximation must be used instead.

2.2 Variational approximation

To assess the sensitivity of a procedure in practice, we need to consider the approximate Bayesian inference algorithm used as well. Here we focus on a variational Bayes approximation due to Blei and Jordan [2006].

Variational Bayes (VB) posits a class of tractable distributions over the model parameters and chooses the element of this class that minimizes the reverse Kullback-Leibler (KL) divergence to the exact posterior. One approach to apply VB to Dirichlet process stick-breaking models assumes $\nu_{K_{\max}} = 1$ for all distributions in the variational class and some truncation level $K_{\max}$. Let ζ collect the first $K_{\max} - 1$ elements of ν, the first $K_{\max}$ elements of β, and the first $K_{\max}$ elements of z_n across n. In what follows, then, we effectively consider the reverse KL divergence to the posterior marginal $\mathcal{P}(\zeta|x)$. By setting $K_{\max}$ sufficiently large, one can make this truncation as accurate as desired.

Mean-field VB is a particularly popular VB variant where the tractable approximating distributions $\mathcal{Q}$ factorize over the parameters. In our case, then, we consider approximations of the form

$$\mathcal{Q}(\zeta|\eta) = \left(\prod_{k=1}^{K_{\max}-1} \mathcal{Q}(\nu_k|\eta)\right)\left(\prod_{k=1}^{K_{\max}} \mathcal{Q}(\beta_k|\eta)\right)\left(\prod_{n=1}^{N} \mathcal{Q}(z_n|\eta)\right), \tag{2}$$

where $\eta \in \Omega_\eta \subseteq \mathbb{R}^{D_\eta}$ represents *variational parameters* that determine the factors of the $\mathcal{Q}$ distribution. When the observation likelihood $\mathcal{P}(x_n|\beta_k)$ is conditionally conjugate with the component-parameter prior $\mathcal{P}_{\text{base}}(\beta_k)$, no further assumptions are needed on the form of $\mathcal{Q}(\beta_k|\eta)$; one can show that it will take the form of the conjugate exponential family after the KL optimization [Blei et al., 2017]. Similarly, when $\mathcal{P}_{\text{stick}}$ is a beta distribution, no further assumptions are needed on $\mathcal{Q}(\nu_k|\eta)$; it will take a beta form. However, since we will consider non-beta forms of $\mathcal{P}_{\text{stick}}$, we must specify a more generic approximation—one that will work even when conditional conjugacy does not hold. To that end, we first transform the ν_k to a value that is unbounded and then use a Gaussian approximation. Define the logit-transformed stick-breaking proportions $\tilde{\nu}_k$:

$$\tilde{\nu}_k := \log(\nu_k) - \log(1 - \nu_k) \quad \Leftrightarrow \quad \nu_k = \frac{\exp(\tilde{\nu}_k)}{1 + \exp(\tilde{\nu}_k)}.$$

We take $\mathcal{Q}(\tilde{\nu}_k|\eta)$ to be a normal distribution, which induces a logit-normal distribution on ν_k. We approximate all resulting integrals over $\mathcal{Q}(\tilde{\nu}_k|\eta)$, as in the KL objective for VB or in our later sensitivity calculations, with Gauss-Hermite (GH) quadrature; see Appendix D.4.

GH quadrature yields an approximation, which we call $\mathrm{KL}(\eta)$, to the full KL, $\mathrm{KL}(\mathcal{Q}(\zeta|\eta)||\mathcal{P}(\zeta|x))$. We minimize that approximation to perform approximate posterior inference:

$$\mathrm{KL}(\mathcal{Q}(\zeta|\eta)||\mathcal{P}(\zeta|x)) = \mathop{\mathbb{E}}_{\mathcal{Q}(\zeta|\eta)}[\log \mathcal{Q}(\zeta|\eta) - \log \mathcal{P}(x,\zeta)] + \log \mathcal{P}(x) \tag{3}$$

$$\hat{\eta} := \mathop{\mathrm{argmin}}_{\eta\in\Omega_\eta} \mathrm{KL}(\eta) \quad \text{where} \quad \mathrm{KL}(\eta) \approx \mathrm{KL}(\mathcal{Q}(\zeta|\eta)||\mathcal{P}(\zeta|x)). \tag{4}$$

Our final approximation to the marginal posterior $\mathcal{P}(\zeta|x)$ is $\mathcal{Q}(\zeta|\hat{\eta})$.

Posterior quantities of interest. To approximate any functional of the exact posterior, we apply the equivalent functional to $\mathcal{Q}(\zeta|\hat{\eta})$. For instance, the approximation to the posterior expected number of clusters among the N observed data points is

$$\mathop{\mathbb{E}}_{\mathcal{Q}(\zeta|\hat{\eta})}[G_{\mathrm{cl}}(z)] = \mathop{\mathbb{E}}_{\mathcal{Q}(z|\hat{\eta})}[G_{\mathrm{cl}}(z)] = \sum_{k=1}^{K_{\max}-1}\left(1 - \prod_{n=1}^{N}(1 - \mathop{\mathbb{E}}_{\mathcal{Q}(z_n|\hat{\eta}_z)}[z_{nk}])\right). \tag{5}$$

We will see examples in Section 7 where our quantity of interest is (a) the expected posterior number of clusters in the observed data, (b) the expected posterior number of clusters in a new set of (as yet unobserved) data, (c) some aspect of a co-clustering matrix, or (d) the topic assignments of certain data points. In all of these cases, as in Eq. 5, we can express our (approximate) posterior quantity of interest as some function g of the optimized variational parameters $\hat{\eta}$: $g(\hat{\eta})$.

Once we have an (approximate) posterior quantity of interest, we can ask how this quantity would change—and whether our substantive scientific conclusions would change—if we had made reasonably different prior choices.

3 A Local Approximation for Sensitivity

We would like to understand how our quantity of interest $g(\hat{\eta})$ changes when the concentration parameter or, more generally, the stick-breaking density $\mathcal{P}_{\mathrm{stick}}$ changes. To efficiently compute these changes, we use a first-order Taylor series approximation in the optimal VB parameters. In this section, we first present the Taylor series and then show how to compute its terms.

Sensitivity to the concentration parameter. First, we show how to approximate the sensitivity of $g(\hat{\eta})$ to the choice of concentration parameter α. Let $\hat{\eta}(\alpha)$ represent the value of $\hat{\eta}$ for a particular choice of α. For our approximation, we choose some initial value α_0 of the concentration parameter and solve the

optimization problem to compute $\hat{\eta}(\alpha_0)$. We then approximate $\hat{\eta}(\alpha)$ with the linear approximation $\hat{\eta}^{\text{lin}}(\alpha)$, and in turn approximate $g(\hat{\eta}(\alpha))$ with $g(\hat{\eta}^{\text{lin}}(\alpha))$:

$$\hat{\eta}^{\text{lin}}(\alpha) := \hat{\eta}(\alpha_0) + \left.\frac{d\hat{\eta}(\alpha)}{d\alpha}\right|_{\alpha_0} (\alpha - \alpha_0) \quad \text{and} \quad g(\hat{\eta}(\alpha)) \approx g(\hat{\eta}^{\text{lin}}(\alpha)). \tag{6}$$

If $\alpha \mapsto \hat{\eta}(\alpha)$ is continuously differentiable, and g is sufficiently smooth, then we expect $g(\hat{\eta}(\alpha)) \approx g(\hat{\eta}^{\text{lin}}(\alpha))$ when $|\alpha - \alpha_0|$ is small. We will show in Theorem 1 below that the map $\alpha \mapsto \hat{\eta}(\alpha)$ is continuously differentiable for our chosen VB approximation.

Sensitivity to the stick-breaking density. Next, we show how to approximate the sensitivity of $g(\hat{\eta})$ to the choice of concentration stick distribution $\mathcal{P}_{\text{stick}}$. Technically, perturbations of α are perturbations of $\mathcal{P}_{\text{stick}}$. But here we consider more general perturbations of the form of $\mathcal{P}_{\text{stick}}$, potentially outside the beta class. To define our perturbations, let $\tilde{\mathcal{P}}$ represent a potentially unnormalized (but normalizable) density with respect to Lebesgue measure; the same notation without the tilde will give the normalized density. Now start from an initial setting of $\mathcal{P}_{\text{stick}}$ at $\mathcal{P}_0$; we will typically start from Dirichlet-process stick-breaking; i.e., $\mathcal{P}_0 = \text{Beta}\,(1, \alpha_0)$ for some α_0. Then take any Lebesgue-measurable function $\phi(\cdot)$ on $[0, 1]$. We consider a range of alternative (potentially unnormalized) stick-breaking forms $\tilde{\mathcal{P}}(\cdot|t)$ defined on $[0, 1]$ by

$$\log \tilde{\mathcal{P}}(\cdot|t) = \log \mathcal{P}_0(\cdot) + t\phi(\cdot). \tag{7}$$

Note that the perturbation applies equally to every stick break ν_k. This style of multiplicative functional perturbation was proposed by Gustafson [1996a]; we deviate from Gustafson [1996a] by considering VB (rather than MCMC) approximations and by allowing ϕ to take on negative values.

If we now let $\hat{\eta}(t)$ represent the value of $\hat{\eta}$ for a particular choice of $\tilde{\mathcal{P}}(\cdot|t)$, we can form an approximation analogous to Eq. 6:

$$\hat{\eta}^{\text{lin}}(t) := \hat{\eta}(0) + \left.\frac{d\hat{\eta}(t)}{dt}\right|_{t=0} (t - 0) \quad \text{and} \quad g(\hat{\eta}(t)) \approx g(\hat{\eta}^{\text{lin}}(t)). \tag{8}$$

As in the case of expansions with respect to α, Eq. 8 is useful only if the map $t \mapsto \hat{\eta}(t)$ is continuously differentiable for the chosen ϕ. As we will show in Theorem 1 below, a sufficient condition for differentiability is given in terms of the following norm on the perturbation ϕ.

$$\text{Define } \|\phi\|_\infty := \operatorname*{esssup}_{\nu_0 \sim \mathcal{P}_0} |\phi(\nu_0)| \text{ and } \quad \mathcal{B}_\phi(\delta) := \{\phi : \|\phi\|_\infty < \delta\}. \tag{9}$$

The set of priors that arise by considering functional perturbations $\phi \in \mathcal{B}_\phi(\delta)$ live in a multiplicative band around the original prior, $\mathcal{P}_0$, as shown in Figure 1. Theorem 1 below states that $t \mapsto \hat{\eta}(t)$ is continuously differentiable whenever $\|\phi\|_\infty < \infty$. So, for sufficiently smooth g, we expect the approximation Eq. 8 to be good for small t, given a particular choice of ϕ with $\|\phi\|_\infty < \infty$.

The functional perturbation given in Eq. 7 is useful because, if we consider any other distribution $\mathcal{P}_1$ for $\mathcal{P}_{\text{stick}}$, we can continuously warp $\mathcal{P}_0$ to $\mathcal{P}_1$ by setting $\phi(\cdot) = \log\left(\mathcal{P}_1(\cdot)/\mathcal{P}_0(\cdot)\right)$ so long as $\mathcal{P}_1 \ll \mathcal{P}_0$; i.e., $\mathcal{P}_1$ is absolutely continuous with respect to $\mathcal{P}_0$. We will see in Section 4 that we can compute an *influence function* to provide an interpretable summary of the effect of arbitrary changes ϕ. Using the influence function and the $\|\cdot\|_\infty$ norm, we are able to find a worst-case choice of ϕ in $\mathcal{B}_\phi(\delta)$.

However, we note that restricting to $\|\phi\|_\infty < \infty$ limits the kinds of alternative priors $\mathcal{P}_1$ that can be formed using Eq. 7. Although we show in Lemma 1 of Appendix A.3 that functional perturbations with $\|\phi\|_\infty < \infty$ yield valid priors, the converse is not true: there exist valid priors $\mathcal{P}_1$ such that the corresponding $\|\phi\|_\infty = \infty$. For instance, perturbing the beta stick-breaking form by changing α provides a counterexample since the log of the beta density is unbounded below; see Example 3 of Appendix A.3 for more details. The limited expressiveness of $\mathcal{B}_\phi(\delta)$ may at first seem like a shortcoming of the perturbation given by Eq. 7. However, we show in Section 4 that, among a class of potential functional perturbations such as those proposed by Gustafson [1996a], only the one we defined in Eq. 7 is *Fréchet differentiable*—and thus can be used to safely reason about worst-case ϕ.

Computing the terms in the Taylor series. It remains to show that $\alpha \mapsto \hat{\eta}(\alpha)$ and $t \mapsto \hat{\eta}(t)$ are continuously differentiable, and to provide a computable formula for the derivative. Differentiability naturally requires some regularity conditions on the VB parameterization and on the optimum. We state sufficient conditions in the following Assumption 1, which is satisfied for any local optimum of a smooth, unconstrained parameterization of the variational approximation.

Assumption 1. *Assume that: (1) the map $\eta \mapsto \mathrm{KL}(\eta)$ is twice continuously differentiable at $\hat{\eta}$; (2) the Hessian matrix $\frac{\partial^2 \mathrm{KL}(\eta)}{\partial \eta \partial \eta^T}\Big|_{\hat{\eta}}$ is non-singular; and (3) there exists an open ball $\mathcal{B}_\eta \subseteq \mathbb{R}^{D_\eta}$ such that $\hat{\eta} \in \mathcal{B}_\eta \subseteq \Omega_\eta$.*

Our next result establishes the differentiability of $\hat{\eta}$ and provides a computable formula for the derivative.

Theorem 1. *Let Assumption 1 hold for the VB approximation given in Section 2.2. Either take $\varepsilon = t$ under the perturbation given by $\log \tilde{\mathcal{P}}(\nu_k|t) = \log \mathcal{P}_0(\nu_k) + t\phi(\nu_k)$ with $\|\phi\|_\infty < \infty$, or take $\varepsilon = \alpha - \alpha_0$ in a perturbation to the concentration parameter α of the unnormalized beta distribution $\log \tilde{\mathcal{P}}(\nu_k|\alpha) = \alpha \log(1-\nu_k)$. Then the map*

$\varepsilon \mapsto \hat{\eta}(\varepsilon)$ is continuously differentiable at $\varepsilon = 0$ with derivative

$$\left.\frac{d\hat{\eta}(\varepsilon)}{d\varepsilon}\right|_{\varepsilon=0} = -\hat{H}^{-1}\hat{J}, \quad \text{where} \quad \rho_k(\nu_k) := \left.\frac{\partial \log \tilde{\mathcal{P}}(\nu_k|\varepsilon)}{\partial \varepsilon}\right|_{\varepsilon=0}, \tag{10}$$

$$\hat{H} := \left.\frac{\partial^2 \mathrm{KL}\left(\eta\right)}{\partial\eta\partial\eta^T}\right|_{\eta=\hat{\eta}}, \quad \mathcal{S}(\zeta|\eta) := \left.\frac{\partial \log \mathcal{Q}(\zeta|\eta)}{\partial\eta}\right|_{\eta}, \text{ and} \tag{11}$$

$$\hat{J} := \frac{\partial}{\partial\eta} \mathop{\mathbb{E}}_{\mathcal{Q}(\zeta|\eta)}\left[\sum_{k=1}^{K_{\max}-1} \rho_k(\nu_k)\right]\Bigg|_{\eta=\hat{\eta}} = \mathop{\mathbb{E}}_{\mathcal{Q}(\zeta|\hat{\eta})}\left[\mathcal{S}(\zeta|\hat{\eta}) \sum_{k=1}^{K_{\max}-1} \rho_k(\nu_k)\right]. \tag{12}$$

Proof. The result follows from Theorem 4 of Appendix A.1, which states general conditions for the differentiability of VB optima. We show in Appendices A.2 and A.3 that the conditions of Theorem 4 are satisfied in the case of our present BNP problem. The equivalence of the expressions for $\hat{J}$ follows by differentiating through the expectation; see Lemma 3 of Appendix B for more details. □

Eq. 10 requires computation of two terms: $\hat{H}^{-1}$ and $\hat{J}$. Typically, $\hat{J}$, which is a derivative of a variational expectation, is straightforward to evaluate: the requisite expectation is evaluated either in closed form or approximated numerically; then, in either case, an application of automatic differentiation provides the gradient [Baydin et al., 2018]. Forming and inverting or factorizing $\hat{H}$ can present a challenge due to its high dimensionality—it has dimensions $D_\eta \times D_\eta$, where D_η is the dimension of η. However, in many cases—including the BNP problem that is our focus—we can take advantage of model sparsity to efficiently compute Eq. 10 (see Section 6), and our experiments confirm that we can compute $\left.\frac{d\hat{\eta}(\varepsilon)}{d\varepsilon}\right|_{\varepsilon=0}$ much more efficiently than re-optimizing the VB objective directly (Section 7.4). Moreover, the savings increases dramatically when we are interested in a range of ε values because $\left.\frac{d\hat{\eta}(\varepsilon)}{d\varepsilon}\right|_{\varepsilon=0}$ can be re-used to for any chosen value of ε.

4 The Influence Function and Worst-Case Functional Perturbations

We next show how to find influential and worst-case functional perturbations to the stick-breaking density. We start by showing how to compute an influence function to summarize the effect of different choices of ϕ. Using the influence function, we are able to design stick-breaking densities that produce a large change in a quantity of interest, including computing the worst-case perturbation in $\mathcal{B}_\phi(\delta)$. To justify such uses of the influence function, we prove that, for multiplicative perturbations and the ∞-norm, the VB objective is Fréchet differentiable—i.e., that it admits a uniformly good linear approximation in a neighborhood of the null perturbation. Finally, we show that our Fréchet differentiability result is unique among a broad class of alternative choices of functional perturbation.

The influence function and worst-case perturbations. We begin by defining the influence function Ψ and discussing its usefulness for understanding the effect of functional perturbations ϕ. Suppose we have a one-dimensional, differentiable quantity of interest, $g(\cdot) : \Omega_\eta \mapsto \mathbb{R}$, and are considering various alternative priors as given by ϕ in Eq. 7. Under the approximation in Eq. 8, the dependence of $g(\hat{\eta}^{\text{lin}}(t))$ on ϕ is not simple if $g(\cdot)$ is non-linear. However, for a particular choice of ϕ, by applying the chain rule with Theorem 1, we can derive a fully linear approximation $g(\hat{\eta}(t)) \approx g(\hat{\eta}) + \frac{dg(\hat{\eta}(t))}{dt}\Big|_{t=0}(t-0)$. The advantage of linearizing g in this way is that the map $\phi \mapsto \frac{dg(\hat{\eta}(t))}{dt}\Big|_{t=0}$ has a particularly simple form, as given by the following result.

Corollary 1. *Under the conditions of Theorem 1, using Eq. 7 with* $\|\phi\|_\infty < \infty$ *and* $\varepsilon = t$*, let* $g(\cdot) : \Omega_\eta \mapsto \mathbb{R}$ *denote a continuously differentiable, real-valued function of interest. Define the* influence function $\Psi : [0,1] \mapsto \mathbb{R}$*:*

$$\Psi(\cdot) := -\sum_{k=1}^{K_{\max}-1} \frac{dg(\eta)}{d\eta^T}\Big|_{\hat{\eta}} \hat{H}^{-1}\mathcal{S}_k(\cdot|\hat{\eta})\mathcal{Q}_k(\cdot|\hat{\eta}), \tag{13}$$

where $\mathcal{S}_k(\cdot|\hat{\eta})$ *and* $\mathcal{Q}_k(\cdot|\hat{\eta})$ *replace* $\mathcal{Q}(\zeta|\eta)$ *with just the factor of* $\mathcal{Q}$ *for* ν_k*. Then the derivative in Eq. 10 can be written as*

$$\frac{dg(\hat{\eta}(t))}{dt}\Big|_0 = \int_0^1 \Psi(\nu_0)\phi(\nu_0)d\nu_0. \tag{14}$$

Proof. The form of the influence function is given applying the chain rule, gathering terms in Eq. 10, and re-writing the variational expectation as an integral over $[0,1]$. We establish an analogous general result for general VB approximations in Corollary 3 of Appendix A.3, specializing to the BNP case in Example 4 of Appendix A.3. □

By choosing perturbations ϕ that align with the influence function, we can form priors that we expect to be influential for the function of interest, $g(\cdot)$. For example, in our experiments of Section 7, we show that by choosing ϕ to be a Gaussian bump aligned with particularly high-magnitude positive or negative values of the influence function, one can ensure a large positive or negative gradient, and hence a large predicted change.

Further, with Corollary 1 in hand, we can find a closed-form expression for the worst-case choice of $\phi \in \mathcal{B}_\phi(\delta)$, which is essentially a VB analogue to Gustafson [1996a, Result 11].

Corollary 2. *Under the conditions of Corollary 1,*

$$\sup_{\phi \in \mathcal{B}_\phi(\delta)} \frac{dg(\hat{\eta}(t))}{dt}\Big|_0 = \delta \int |\Psi(\nu_0)|\,\mu(d\nu_0),$$

and the supremum is achieved at the perturbation $\phi^*(\cdot) = \delta\,\text{sign}\,(\Psi(\cdot))$*.*

Proof. The result follows immediately from applying Hölder's inequality to Eq. 14. We establish a similar but much more general result for VB approximations with general choices of model and parameters in Corollary 4 of Appendix A.4. The present result is a special case using Example 4 of Appendix A.4. □

In our experiments of Section 7, we use Corollaries 1 and 2 to choose influential pertubations, and then use the partially linearized Eq. 8 to make predictions about the effect of the pertubations.

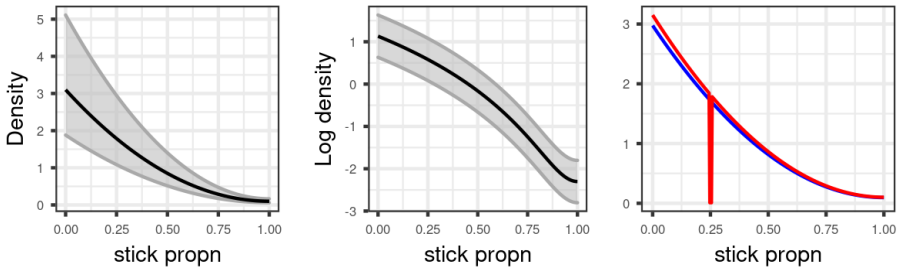


Figure 1: Left two: A multiplicative ball $\mathcal{B}_\phi(\delta)$. Right: Two densities that are distant according to reverse KL divergence and $\|\cdot\|_\infty$ but close according to $\|\cdot\|_p$ for $p \in [1, \infty)$.

Multiplicative perturbations are continuously Fréchet differentiable. The influence function provides a succinct summary of the effect of all perturbations $\phi \in \mathcal{B}_\phi(\delta)$, which we might hope to be accurate for sufficiently small δ. However, the accuracy of our approximation within $\mathcal{B}_\phi(\delta)$ is not guaranteed by Theorem 1 alone. Specifically, Theorem 1 states only that, for a *particular* direction ϕ, $t \mapsto \hat{\eta}(t)$ is continuously differentiable—i.e. that, for a fixed ϕ, one can make t sufficiently small so that the error $\left|\hat{\eta}(t) - \hat{\eta}^{\text{lin}}(t)\right|$ goes to zero faster than t. But, if we write $\hat{\eta}(t\phi)$ and $\hat{\eta}^{\text{lin}}(t\phi)$ to make the dependence on ϕ explicit, then Theorem 1 does not imply that for a fixed δ (no matter how small), the worst-case error $\sup_{\phi \in \mathcal{B}_\phi(\delta)} \left|\hat{\eta}(\phi) - \hat{\eta}^{\text{lin}}(\phi)\right|$ is bounded, much less that it goes to zero faster than δ.

Thus, to be assured that the influence function is a meaningful summary of the effect of all $\phi \in \mathcal{B}_\phi(\delta)$, we wish to establish that the linear approximation given by Eq. 8 is uniformly accurate over all ϕ of interest within a sufficiently small neighborhood of the zero function. Specifically, observing that ϕ is a point in the Banach space L_∞ [Dudley, 2018, Theorem 5.2.1], we wish to establish that the map $\phi \mapsto \hat{\eta}(\phi)$ from L_∞ to $\mathbb{R}^{D_\eta}$ is *Fréchet differentiable*, as we now formally define.

Definition 1. (Fréchet differentiability, [Zeidler, 1986, Definition 4.5]) Let B_1 and B_2 denote Banach spaces, and let $\mathcal{B}_1 \subseteq B_1$ define an open neighborhood of $\phi_0 \in B_1$. A function $f : \mathcal{B}_1 \mapsto B_2$ is *Fréchet differentiable* at ϕ_0 if there exists a

bounded linear operator, $f^{\text{lin}} : B_1 \mapsto B_2$, such that, for $\phi \in B_1$,

$$f(\phi) - f(\phi_0) - f^{\text{lin}}_{\phi_0}(\phi - \phi_0) = o(\|\phi - \phi_0\|) \quad \text{as} \quad \|\phi - \phi_0\| \to 0.$$

The function f is *continuously Fréchet differentiable* if the map $\phi_0 \mapsto f^{\text{lin}}_{\phi_0}(\cdot)$ is continuous as a map from from $\mathcal{B}_1$ to the space of all continuous linear operators from B_1 to B_2 equipped with the operator norm. □

By Zeidler [1986, Proposition 4.8], if a function is Fréchet differentiable, then the linear operator f^{lin} is given precisely by the directional derivative $df(\phi_0 + t(\phi - \phi_0))/dt$. Thus, if $\phi \mapsto \hat{\eta}(\phi)$ is Fréchet differentiable, its derivative is given by Corollary 1. Fréchet differentiability guarantees that, for sufficiently small δ, the error of the linear approximation given by Corollary 1 does not blow up in the ball $\mathcal{B}_\phi(\delta)$.

We emphasize that Fréchet differentiability is neither sufficient nor necessary for a derivative to be useful. For example, it is possible in principle for a function to be Fréchet differentiable but still have a very large finite second derivative, and so fail to extrapolate meaningfully to any alternatives one cares about. Conversely, if a function fails to be Fréchet differentiable, the derivative may still perform well in particular directions, including that chosen by Corollary 2. Nevertheless, Fréchet differentiability is a strong local result, and provides some assurance that one can use results such as Corollary 2 without uncovering pathological behavior.

Finally, then, we prove that our perturbation is continuously Fréchet differentiable.

Theorem 2. *Under the conditions of Theorem 1, the map $\phi \mapsto \hat{\eta}(\phi)$ is well-defined and continuously Fréchet differentiable in a neighborhood of the zero function as a map from L_∞ to $\mathbb{R}^{D_\eta}$, with the derivative given in Corollary 1.*

Proof. Our result here is a special case of our general result for VB approximations given in Theorem 5 of Appendix A.4. □

Many other functional perturbations and norms are not Fréchet differentiable. So far we have focused on the multiplicative functional perturbations in Eq. 7 combined with the infinity norm in Eq. 9. We now ask whether we could perform a similar analysis for other functional perturbations. We show that, of the perturbations proposed by Gustafson [1996a], only multiplicative perturbations yield Fréchet differentiable VB optima.

Specifically, Gustafson [1996a] examines general perturbations, from initial prior $\mathcal{P}_0$ to alternative $\mathcal{P}_1$, that take the following form—with θ a parameter $\theta \in \Omega_\theta \subseteq \mathbb{R}^{D_\theta}$ and $p \in [1, \infty)$:

$$\tilde{\mathcal{P}}(\theta|t_p) := \left((1 - t_p)\mathcal{P}_0(\theta)^{1/p} + t_p \frac{1}{p}\mathcal{P}_1(\theta)^{1/p}\right)^p. \tag{15}$$

Again, let ϕ represent the perturbation, now with:

$$\phi(\theta|\mathcal{P}_1, p) := \mathcal{P}_1(\theta)^{1/p} - \mathcal{P}_0(\theta)^{1/p} \quad \text{and} \quad \|\phi\|_p := \left(\int_0^1 |\phi(\theta)|^p \, d\theta\right)^{1/p}. \tag{16}$$

The limit $p \to \infty$ recovers our multiplicative perturbation in Eq. 7 with infinity norm in Eq. 9. The choice $p = 1$ recovers a purely additive perturbation. Gustafson [1996a, Result 2] states that $\|\phi\|_p < \infty$ ensures that the corresponding $\tilde{\mathcal{P}}(\theta|t_p)$ can be normalized, strongly motivating using the $\|\cdot\|_p$ norm with the perturbation given by Eq. 15.

Our next theorem shows that the reverse KL divergence is discontinuous in $\|\cdot\|_p$ for $p < \infty$. Since Fréchet differentiability implies continuity [Zeidler, 1986, Proposition 4.8 (d)], Theorem 3 implies that it is impossible to derive an analogue of Theorem 2 for perturbations of the form in Eq. 15 with the norms in Eq. 16.

Theorem 3. *Let μ denote a measure on the Borel sets of some domain Ω_θ, with μ absolutely continuous with respect to the Lebesgue measure, and let $\mathcal{Q}(\theta)$ and $\mathcal{P}_0(\theta)$ denote densities with respect to μ. Without loss of generality, assume that $\mathcal{Q}(\theta) > 0$ on Ω_θ. Assume that $\mathrm{KL}\left(\mathcal{Q}(\theta)||\mathcal{P}_0(\theta)\right)$ is well-defined and finite.*

Then, for any $\epsilon > 0$ and any $M > 0$, we can find a density $\mathcal{P}_1(\theta)$ such that $\|\phi(\theta|\mathcal{P}_1, p)\|_p < \epsilon$ but $|\mathrm{KL}\left(q(\theta)||\mathcal{P}_1(\theta)\right) - \mathrm{KL}\left(q(\theta)||\mathcal{P}_0(\theta)\right)| > M$.

Proof. See Appendix A.5 for a constructive proof, the key to which is the fact that in any $\|\cdot\|_p$ neighborhood of zero there exist prior densities taking values arbitrarily close to zero on sets of nonzero measure, for which the reverse KL divergence blows up. □

Recall from Section 3 (and particularly Example 3 of Appendix A.3) that there exist priors that cannot be formed from Eq. 7 using ϕ with $\|\phi\|_\infty < \infty$. In light of the proof of Theorem 3, the limited expressiveness of multiplicative perturbations with the $\|\cdot\|_\infty$ norm looks like a feature rather than a bug. Consider the rightmost panel of Figure 1, which illustrates the tradeoffs between the various norms. The two blue and red densities are far from one another according to reverse KL divergence since the red density takes values that are nearly zero where the blue density has nonzero mass. The two densities are also distant in $\|\cdot\|_\infty$ since it takes a large multiplicative change to turn the nonzero blue density into the nearly zero red density. However, the two densities are close in $\|\cdot\|_p$ since the region where the red density is nearly zero has a small measure. In order for VB approximations to be continuous (a necessary condition for Fréchet differentiability), one must consider a topology on priors that is no coarser than the topology induced by reverse KL divergence. But since valid priors can take values close to zero, a sacrifice in expressiveness of the neighborhood of zero must be made in order to induce a topology that is compatible with reverse KL divergence. Multiplicative changes and the $\|\cdot\|_\infty$ norm implement such a tradeoff in a natural, easy-to-understand way.

In this sense, VB approximations based on reverse KL divergence are inherently non-robust to priors that ablate mass nearly to zero. No parameterization of the space of priors will relieve this non-robustness. Only by basing variational approximations on divergences other than reverse KL might this non-robustness be alleviated.

5 Related Work

Evaluating sensitivity to prior choices is typically a necessary step in applied Bayesian data analysis [Gelman et al., 2013, Chapter 6], and a central aim of Bayesian robustness is to provide methods and metrics to measure sensitivity of posterior quantities to variations in the model [Insua and Ruggeri, 2000]. One family of approaches to robustness quantification, "local robustness," forms differential approximations to model sensitivity, in light of the fact that more desirable "global sensitivity" measures are computationally expensive or infeasible in all but special cases [Sivaganesan, 2000, Gustafson, 2000]. Two recent papers employing non-local approaches to BNP robustness are Nieto-Barajas and Prünster [2009] and Saha and Kurtek [2019], both of which run new MCMC chains at a set of alternative priors. The present work is essentially a VB extension and application of local Bayesian robustness literature which was based on MCMC samples from the full posterior [Gustafson, 1996a,b]. In contrast to MCMC, for which the derivatives of local robustness must be approximated with (potentially noisy) sample covariances, VB optima admit closed-form derivatives. In addition to extending local robustness methods to VB, we contrast with previous applications of local robustness by evaluating the ability of our linear approximation to *extrapolate* to alternative priors, rather than considering the derivative to be measure of robustness *per se* (as in, for example, Basu [2000]).

Since VB is an optimization procedure, the evaluation of the sensitivity of VB estimates inherits a long tradition of robustness methods in frequentist statistics (e.g. [Jaeckel, 1972, Cook, 1986, Hampel et al., 2011]). In particular, our theoretical results extend the results of Giordano et al. [2018], providing more easily verifiable sufficient conditions for Giordano et al. [2018, Theorem 2] and proving results for non-parametric perturbations, including continuous Fréchet differentiability (and non-differentiability).

One of our quantities of interest is the posterior expected number of clusters, a quantity which we find non-robust to prior specification in certain cases (see Sections 7.1 and 7.3 and Appendix E.4). The non-robustness of the posterior expected number of clusters can be seen as a companion result to recent literature showing that BNP models provide inconsistent estimates of the number of clusters when the true data generating process is a finite mixture model (Miller and Harrison [2013, 2014]; Cai et al. [2020, 2021] prove similar results for finite mixture models). As Miller and Harrison [2014] observe empirically, inconsistency can be attributed to a small number of low-probability clusters in the BNP posterior. Similarly, we find that the small, rare clusters account for the non-robustness of the posterior expected number of clusters. Whereas inconsistency says that the posterior number of clusters may be unreliable even as the number of data points tend towards infinity, we show here that with a fixed data set, the number of clusters may be unreliable due to the subjective nature of the prior specification.

6 Fast Computation of the Sensitivity

A principal challenge of computing the sensitivity efficiently is the high-dimensional nature of the parameter ζ and hence the variational parameters η. In particular, we have seen that, in our BNP stick-breaking model, ζ and η both grow linearly with the number of data points N. This growth leads to two major computational challenges: (1) we must solve a high-dimensional optimization problem to extremize the VB objective, and (2) we must solve a linear system given by the Hessian $\hat{H}$. Here we show how we can use special structure in the model to reduce to low-dimensional problems and thereby enjoy efficient computation.

Global and local parameters. In both cases, the key to reducing to a lower-dimensional problem is separating *global* and *local* parameters. Global variables are common to all data points. Local variables are unique to each data point. For instance, in a Gaussian (or other typical) mixture model, the stick-breaking proportions ν and component parameters β are global, whereas the cluster assignment parameters z are local.

Let γ denote the collection of global parameters. When we use a standard mean-field VB parameterization, the VB distributions on γ have their own variational parameters, which we denote η_γ. Similarly, let ℓ denote the local parameters and let η_ℓ be the corresponding local variational parameters.

Reducing to optimization over the global variational parameters. We next show how to reduce the potentially high-dimensional optimization problem over all of η to optimizing over just the global variational parameters η_γ.

In all models we will consider, the conditional posterior $\mathcal{P}(z|\gamma, x)$ has a tractable closed form. Since we choose a conjugate mean-field approximating family for $\mathcal{Q}(z|\eta)$, the optimal local variational parameters $\hat{\eta}_\ell$ can be written as a closed-form function of the global variational parameters η_γ. For some prior parameter ε (as in Theorem 1), let $\hat{\eta}_\ell(\eta_\gamma; \varepsilon)$ denote this mapping, so that

$$\hat{\eta}_\ell(\eta_\gamma; \varepsilon) := \operatorname*{argmin}_{\eta_\ell} \mathrm{KL}\left((\eta_\gamma, \eta_\ell), \varepsilon\right). \tag{17}$$

In Example 6 (Appendix D.1), we illustrate this technique for a Gaussian mixture model. Using Eq. 17, we can rewrite our objective as a function of the global parameters. Define

$$\mathrm{KL}_{\mathrm{glob}}(\eta_\gamma, \varepsilon) := \mathrm{KL}\Big((\eta_\gamma, \hat{\eta}_\ell(\eta_\gamma; \varepsilon)), \varepsilon\Big).$$

The $\hat{\eta}_\gamma(\varepsilon)$ that minimizes $\mathrm{KL}_{\mathrm{glob}}(\eta_\gamma, \varepsilon)$ is the same as the corresponding sub-vector of the $\hat{\eta}(\varepsilon)$ that minimizes $\mathrm{KL}\left(\eta, \varepsilon\right)$.

Rather than optimizing the $\mathrm{KL}\left(\eta\right)$ over all variational parameters, we numerically optimize $\mathrm{KL}_{\mathrm{glob}}$, which is a function only of the relatively low-dimensional global parameters. To minimize $\mathrm{KL}_{\mathrm{glob}}(\eta_\gamma)$ in practice, we run the BFGS algorithm with a loose convergence tolerance followed by the trust-region Newton conjugate gradient method to find a high-quality optimum (the `trust-ncg` method of `scipy.optimize.minimize`, Virtanen et al. [2020]; see also Nocedal and Wright

[2006, Chapter 7]). After the optimization terminates at an optimal $\hat{\eta}_\gamma$, the optimal local parameters $\hat{\eta}_\ell$ can be set in closed form to produce the entire vector of optimal variational parameters, $\hat{\eta} = (\hat{\eta}_\gamma, \hat{\eta}_\ell)$.

6.1 Computing and inverting the Hessian

Since the dimension D_η of η scales with N, we can quickly reach cases where inverting or even instantiating a dense matrix of size $D_\eta \times D_\eta$ in memory would be prohibitive. The key to efficient computation is that $\hat{H}$ is not dense; we will again exploit structure inherent in the global/local decomposition.

For generic variables a and b, let H_{ab} denote the sub-matrix $\partial^2 \mathrm{KL}(\eta) / \partial \eta_a \eta_b^T \big|_{\hat{\eta}}$, the Hessian with respect to the variational parameters governing a and b. We decompose the Hessian matrix $\hat{H}$ into four blocks according to the global/local decomposition:

$$\hat{H} = \left.\frac{\partial^2 \mathrm{KL}(\eta)}{\partial \eta \partial \eta^T}\right|_{\hat{\eta}} = \begin{pmatrix} H_{\gamma\gamma} & H_{\gamma\ell} \\ H_{\ell\gamma} & H_{\ell\ell} \end{pmatrix}.$$

Similarly, let $\hat{J}_\gamma$ be the components of $\hat{J}$ corresponding to the variational parameters η_γ. The local components, $\hat{J}_\ell$, are zero since no local variables enter the expectation in Eq. 12 when we are perturbing the stick-breaking distribution.

In this notation,

$$\left.\frac{d\hat{\eta}(t)}{dt}\right|_{t=0} = -\begin{pmatrix} H_{\gamma\gamma} & H_{\gamma\ell} \\ H_{\ell\gamma} & H_{\ell\ell} \end{pmatrix}^{-1} \begin{pmatrix} \hat{J}_\gamma \\ 0 \end{pmatrix}. \tag{18}$$

Applying the Schur complement and focusing on the global parameters (see Appendix D.2 for more details), we find

$$\left.\frac{d\hat{\eta}_\gamma(t)}{dt}\right|_{t=0} = -\hat{H}_\gamma^{-1}\hat{J}_\gamma \quad \text{where} \quad \hat{H}_\gamma := \left(H_{\gamma\gamma} - H_{\gamma\ell}H_{\ell\ell}^{-1}H_{\ell\gamma}\right), \tag{19}$$

In our model, $H_{\ell\ell}$ is block diagonal, and the size of $H_{\gamma\gamma}$ is relatively small. Thus each term of Eq. 19 can be tractably computed, even on very large datasets. While the Schur complement calculation is illustrative, Eq. 19 is equivalent to applying automatic differentiation to the global-only objective $\mathrm{KL}_{\mathrm{glob}}(\eta_\gamma, t)$; see Appendix D.2 for details.

In our BNP applications, it is not cost-effective to form and invert or factorize $\hat{H}$ in memory. Instead, we numerically solve linear systems of the form $\hat{H}^{-1}v$ using the conjugate gradient (CG) algorithm [Nocedal and Wright, 2006, Chapter 5], which requires only Hessian-vector products that are readily available through automatic differentiation.

A linear approximation only in the global variational parameters. With the tools above, we can separate out the linear approximation in the global

parameters and then directly compute the local parameters. In particular, we compute

$$\hat{\eta}_\gamma^{\text{lin}}(t) := \hat{\eta}_\gamma + \left.\frac{d\hat{\eta}_\gamma(t)}{dt}\right|_{t=0} t, \tag{20}$$

and then use $\hat{\eta}_\ell(\eta_\gamma)$ e.g. in computing our quantity of interest. By doing so, our approximation is able to retain non-linearities in the map $\eta_\gamma \mapsto \hat{\eta}_\ell(\eta_\gamma)$. We give an example for the expected number of clusters in Appendix D.3. In all our experiments, we use Eq. 20 in this way.

7 Experimental Results

We next evaluate our sensitivity approximations on three real data sets, each with a different model using stick-breaking. We find that our approximations largely agree with ground truth obtained by re-running the VB optimization, but with the evaluation of our derivative an order of magnitude faster than re-optimizing for a given perturbation.

7.1 Gaussian mixture modeling on iris data

We perform a clustering analysis of Fisher's iris data set [Anderson, 1936]. Here each data point (with $N = 150$ total points) represents $d = 4$ measurements of a particular flower, from one of three iris species. We use a standard Gaussian mixture model with a conjugate Gaussian-Wishart prior for the component parameters (detailed in Appendix E.2) and a mean-field VB approximation with truncation parameter $K_{\max} = 15$. We consider two quantities of interest: (1) g_{cl}, the posterior expected number of clusters among the N observed data points, and (2) $g_{\text{pred,cl}}$, the posterior predictive expected number of clusters in N new (i.e. as-yet-unseen) data points. We set the base stick-breaking prior $\mathcal{P}_0(\nu_k)$ to be the standard Beta $(\nu_k|1, \alpha)$ distribution with $\alpha = \alpha_0 = 2$. Under the base stick-breaking prior with α_0, the posterior expected number of clusters matches the three iris species; see also Figure 13 in Appendix E.2 for an illustration.

Sensitivity to the concentration parameter. We approximate the changes in the quantities of interest as α varies over $\alpha \in [0.1, 4.0]$, which corresponds to an *a priori* expected number of clusters among N data points in $[1.5, 15]$ (Appendix E.1). Over this range, the shape of a Beta $(1, \alpha)$ density varies considerably, as shown in Figure 12 in Appendix E.1.

Figure 2 compares our linear approximation to ground truth on the two quantities of interest as α varies. Over this range of α, the posterior expected number of clusters in the observed data is quite robust; it remains nearly constant at three. The posterior predictive expected number of clusters in N new data points is less robust; it ranges roughly from 3.0 to 5.6 expected species. Our approximation captures this qualitative behavior. As expected, the approximation is least accurate furthest from the α_0, where the Taylor series is centered.

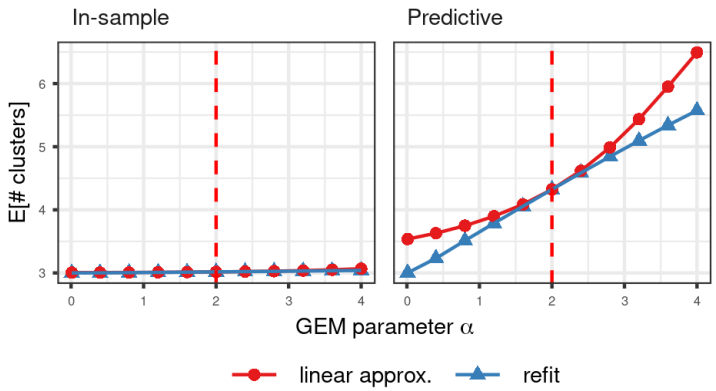


Figure 2: The expected number of clusters in the original data set (g_{cl}, left) and in a new data set of size N ($g_{\text{pred,cl}}$, right) as α varies in the GMM fit of the iris data. We formed the linear approximation at $\alpha_0 = 2$.

Sensitivity to functional perturbations. Insensitivity of the expected number of clusters g_{cl} to α does not rule out sensitivity to other prior perturbations. We now check how our approximation fares for the multiplicative perturbations in Eq. 7. We consider perturbations ϕ that are Gaussian bumps in logit stick space, with each perturbation centered at a different location on the real line. Each row of Figure 3 corresponds to a different ϕ. Each ϕ is shown in gray in the leftmost plot of its row. The middle column of Figure 3 shows the stick-breaking prior $\mathcal{P}(\nu_k|\phi)$ induced by the corresponding ϕ. The rightmost column of Figure 3 shows the changes produced by the ϕ perturbation for that row. We see that our approximation captures the qualitative behavior of the exact changes.

We also see in this example that we can use the influence function to predict the effect of functional changes to the stick-breaking prior. In the leftmost column, we plot in purple the influence function in the logit space.[1] According to Corollary 1, the sign and magnitude of the effect of a perturbation should be determined by its integral against the influence function. Thus, when ϕ lines up with a negative part of Ψ, as in the first row, we expect the change to be negative. Similarly, we expect the perturbation of the bottom row to produce a positive change, and the middle row, in which ϕ overlaps with both negative and positive parts of the influence function, to produce a relatively small change. We see this intuition borne out in the rightmost column.

Worst-case functional perturbation. Finally, Figure 4 shows the worst-case multiplicative perturbation with $\|\phi\|_\infty = 1$, as given by Corollary 2, along with its effect on the prior and g_{cl}. As expected, this worst-case perturbation has a much

[1] Corollary 1 expresses the influence function in the stick domain $[0, 1]$, but, for visualization, it is preferable to express the influence function in the logit stick domain $\mathbb{R}$. The more general Corollary 3 in Appendix A.3 accomodates such transformations.

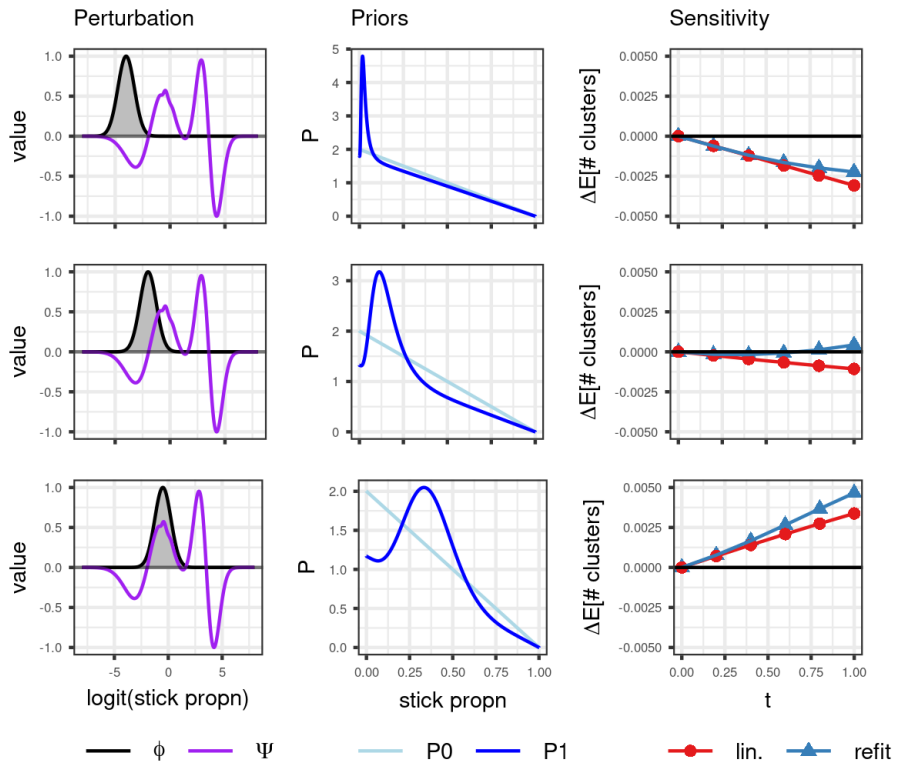


Figure 3: Sensitivity of the expected number of in-sample clusters in the iris data set to three multiplicative perturbations each with $\|\phi\|_\infty = 1$. (Left) The multiplicative perturbation ϕ is in grey. The influence function Ψ, scaled so $\|\Psi\|_\infty = 1$, is in purple. (Middle) The initial $\mathcal{P}_0(\nu_k)$ (light blue) and alternative $\mathcal{P}_1(\nu_k)$ (dark blue) priors. (Right) The effect of the perturbation on the change in expected number of in-sample clusters for $t \in [0, 1]$.

larger effect on g_{cl} compared to the other unit-norm perturbations in Figure 3. However, even with the worst-case perturbation—which results in an unreasonably shaped prior density—the change in g_{cl} is still small. We conclude that g_{cl} appears to be a robust quantity for this model and dataset.

7.2 Regression mixture modeling

We next check our approximation on a more complex clustering task: clustering time series, with a co-clustering matrix (and summaries thereof) as the quantity of interest.

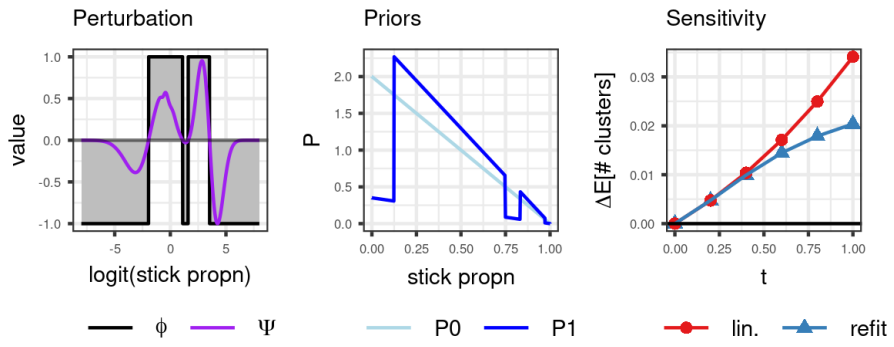


Figure 4: Sensitivity of the expected number of in-sample clusters in the iris data set to the worst-case multiplicative perturbation with $\|\phi\|_\infty = 1$.

Data and model. We use a publicly available data set of mice gene expression [Shoemaker et al., 2015]. Mice were infected with influenza virus, and expression levels of a set of genes were assessed at 14 time points after infection. Three measurements were taken at each time point (called biological replicates), for a total of $M = 42$ measurements per gene.

The goal of the analysis is to cluster the time-course gene expression data under the assumption that genes with similar time-course behavior may have similar function. Clustering gene expressions is often used for exploratory analysis and is a first step before further downstream investigation. It is important, therefore, to ascertain the stability of the discovered clusters.

The left plot of Figure 14 in Appendix E.3 shows the measurements of a single gene over time. We model each gene as belonging to a latent component, where each component defines a smooth expression curve over time. Then, observations are drawn by adding i.i.d. noise to the smoothed curve along with a gene-specific offset. Following Luan and Li [2003], we construct the smoothers using cubic B-splines.

Let $x_n \in \mathbb{R}^M$ be measurements of gene n at M time points. Let A be the $M \times d$ B-spline regressor matrix, so that the ij-th entry of A is the j-th B-spline basis vector evaluated at the i-th time point. The right plot of Figure 14 in Appendix E.3 shows the B-spline basis. The distribution of the data arising from component k is

$$\mathcal{P}(x_n|\beta_k, b_n) = \mathcal{N}\left(x_n|A\mu_k + b_n, \tau_k^{-1} I_{M\times M}\right), \tag{21}$$

where b_n is a gene-specific additive offset and I is the identity matrix. We include the additive offset because we are interested in clustering gene expressions based on their patterns over time, not their absolute level. In this model, the component-specific parameters are $\beta_k = (\mu_k, \tau_k)$, the regression coefficients and the inverse noise variance. The component frequencies are determined by stick-breaking

according to ν, and cluster assignments z are drawn as in Section 2.1.

Our variational approximation factorizes similarly to Eq. 2 except with an additional factor for the additive shift. In our variational approximation, we also make a simplification by letting $\mathcal{Q}(\beta_k|\eta) = \delta(\beta_k|\eta)$, where $\delta(\cdot|\eta)$ denotes a point mass at a parameterized location. See Appendix E.3 for further details concerning the model and variational approximation.

Quantity of interest: the co-clustering matrix and summaries. In this application, we are particularly interested in which genes cluster together, so we focus on the posterior co-clustering matrix. Let $g_{\text{cc}}(\eta) \in \mathbb{R}^{N\times N}$ denote the matrix whose (i,j)-th entry is the posterior probability that gene i belongs to the same cluster as gene j, given by

$$[g_{\text{cc}}(\eta)]_{ij} = \mathop{\mathbb{E}}_{\mathcal{Q}(z|\eta)}[\mathbb{I}(z_i = z_j)] = \begin{cases} \sum_{k=1}^{K_{\max}} \left(\mathop{\mathbb{E}}_{\mathcal{Q}(z_i|\eta)}[z_{ik}] \mathop{\mathbb{E}}_{\mathcal{Q}(z_j|\eta)}[z_{jk}] \right) & \text{for } i \neq j \\ 1 & \text{for } i = j. \end{cases}$$

Figure 5 shows the inferred co-clustering matrix at α_0.

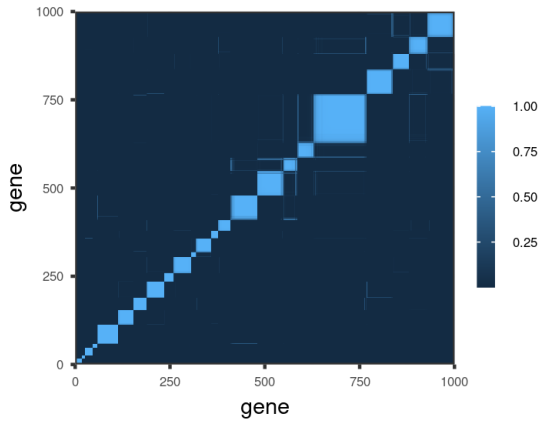


Figure 5: The inferred co-clustering matrix of gene expressions at $\alpha_0 = 6$.

Below, we will use the influence function (Corollary 2) to try and find a perturbation that produces large changes in the co-clustering matrix. To compute the worst-case perturbation, we must choose a univariate summary of the N^2-dimensional co-clustering matrix whose derivative we wish to extremize. We use the sum of the eigenvalues of the symmetrically normalized graph Laplacian, as given by

$$g_{\text{ev}}(\eta) = \text{Tr}\left(I - D(\eta)^{-1/2} g_{\text{cc}}(\eta) D(\eta)^{-1/2}\right),$$

where $D(\eta)^{-1/2}$ is the diagonal matrix with entries $d_i = \sum_{j=1}^{N} [g_{\text{cc}}(\eta)]_{ij}$. The quantity g_{ev} is differentiable, and has close connection with the number of distinct

components in a graph [von Luxburg, 2007]. We expect that prior perturbations that produce large changes in g_{ev} will also produce large changes in the full co-clustering matrix.

Sensitivity to the concentration parameter. We first evaluate the sensitivity of the co-clustering matrix g_{cc} to the choice of α in the stick-breaking prior.

We start at $\alpha = \alpha_0 = 6$. We use the linear approximation to extrapolate the co-clustering matrix under prior parameters $\alpha = 0.1$ and $\alpha = 12$. The *a priori* expected number of clusters in the original data at these values is 2 and 50, respectively. Despite this wide prior range, the change in the posterior co-clustering matrix for each α is minuscule (Figure 6). The largest absolute changes in the co-clustering matrix is of order 10^{-2}. Refitting the approximate posterior at $\alpha = 0.1$ and $\alpha = 12$ confirms the insensitivity predicted by the linearized variational global parameters. Beyond capturing insensitivity, the linearized parameters were also able to capture the sign and size of the changes in the individual entries of the co-clustering matrix, even though these changes are small.

Sensitivity to functional perturbations. We now investigate sensitivity of the co-clustering matrix to deviations from the beta prior. In Figure 7, we use the influence function for g_{ev} to construct a nonparametric prior perturbation that we expect to have a large, positive effect. The resulting prior does indeed produce changes an order of magnitude larger than those produced by the perturbations to α shown in Figure 6, and our approximation is again able to capture the qualitative changes. The influence function is also able to explain why α perturbations were unable to produce large changes in this case: Figure 8 shows that changing α (as in Example 3) induces large changes in the prior only where the influence function is small.

However, even with the (unreasonable-looking) selected functional perturbation, the size of the differences in the co-clustering matrix remains modest. It is unlikely that any scientific conclusions derived from the co-clustering matrix would have changed after the functional perturbation. The co-clustering matrix appears robust to perturbations in the stick-breaking distribution.

7.3 Genetic admixture modeling with fastSTRUCTURE

Our final analysis illustrates the use of our approximation for stick-breaking priors beyond clustering; namely, in topic modeling.

Data and model. We use a publicly available dataset that contains genotypes from $N = 155$ individuals of an endangered bird species, the Taita thrush [Galbusera et al., 2000]. Individuals were collected from four regions in southeast Kenya (Chawia, Mbololo, Ngangao, Yale), and each individual was genotyped at $L = 7$ micro-satellite loci. The four regions were once part of a cohesive cloud forest that has been fragmented by human development. For this endangered bird species, understanding the degree to which populations have grown genetically distinct is important for conservation efforts: well-separated populations with little genetic diversity are particularly at risk of extinction. The goal of the analysis is

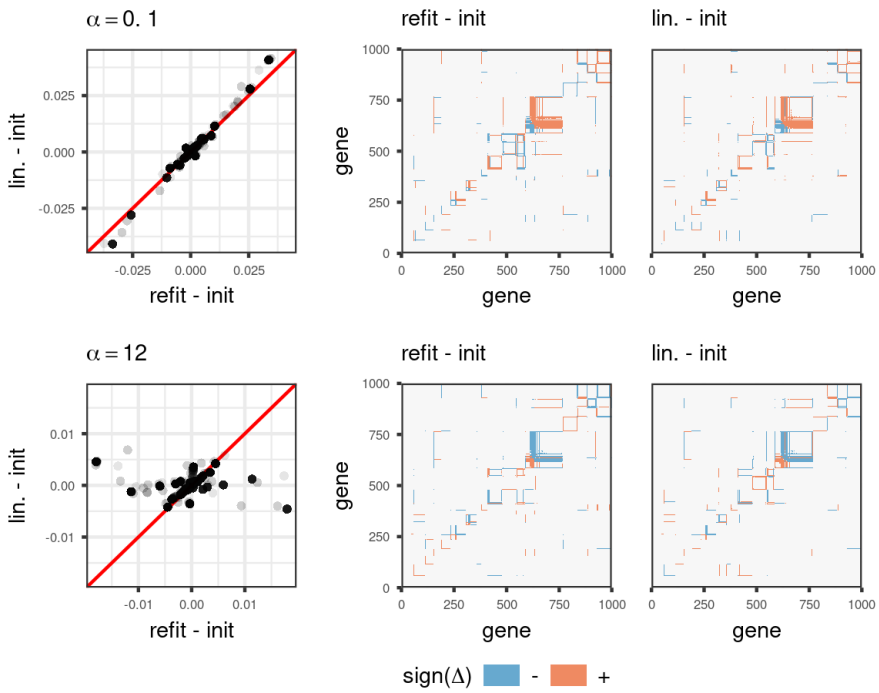


Figure 6: Differences in the co-clustering matrix at $\alpha = 0.1$ (top row) and $\alpha = 12$ (bottom row), relative to the co-clustering matrix at $\alpha_0 = 6$. (Left) A scatter plot of differences under the linear approximation against differences after refitting. Each point represents an entry of the co-clustering matrix. Note the scales of the axes: the largest change in an entry of the co-clustering matrix is ≈ 0.03. (Middle) Sign changes in the co-clustering matrix observed after refitting, ignoring the magnitude of the change. (Right) Sign changes under the linearly approximated variational parameters. For visualization, changes with absolute value $< 10^{-5}$ are not colored.

to infer the population of origin for specific loci and estimate the degree to which populations are admixed in each individual.

Let $x_{nli} \in \{1, \ldots, J_l\}$ be the observed genotype for individual n at locus l and chromosome i. J_l is the number of possible genotypes at locus l. For example, if the measurements are all single nucleotides (A, T, C or G) then $J_l = 4$ for all l.

A latent population is characterized by the collection $\beta_k = (\beta_{k1}, \ldots, \beta_{kL})$, where $\beta_{kl} \in \Delta^{J_l - 1}$ are the latent frequencies for the J_l possible genotypes at locus l. Let z_{nli} be the assignment of observation x_{nli} to a latent population. Notice that for a given individual n, different loci (or even different chromosomes at a given locus) may have different population assignments. The distribution of

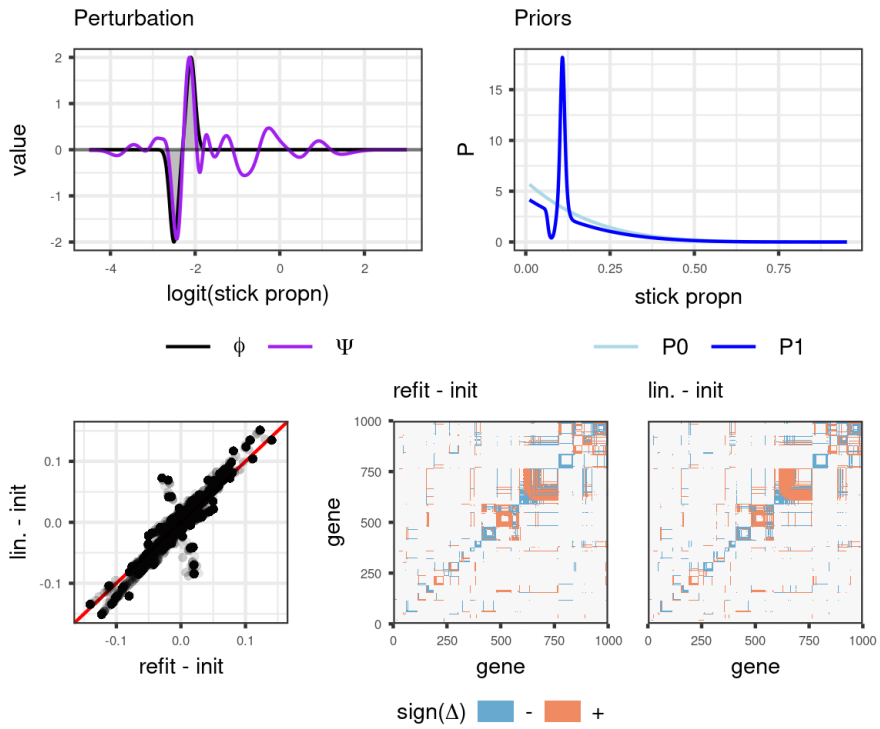


Figure 7: Effect on the co-clustering matrix of a multiplicative functional perturbation. (Top left) The perturbation ϕ is in grey, and the influence function is in purple. (Top right) The effect of this perturbation on the prior density. (Bottom) The effect of this perturbation on the co-clustering matrix. Note the scale of the scatter plot axes compared with the scatter plots in Figure 6.

$x_{nli} \in \{1, \ldots, J_l\}$ arising from population k is $\mathcal{P}(x_{nli}|\beta_k) = \text{Categorical}\,(x_{nli}|\beta_{kl})$.

Unlike the previous models, we now have a stick-breaking process for each individual. Draw sticks

$$\nu_{nk} \overset{indep}{\sim} \mathcal{P}_{\text{stick}}(\nu_{nk}), \quad n = 1, \ldots, N; k = 1, 2, \ldots.$$

The prior assignment probability vector $\pi_n = (\pi_{n1}, \pi_{n2}, \ldots)$, now unique to each individual, is formed by the same stick-breaking construction as before,

$$\pi_{nk} = \nu_{nk} \prod_{k'<k} (1 - \nu_{nk'}).$$

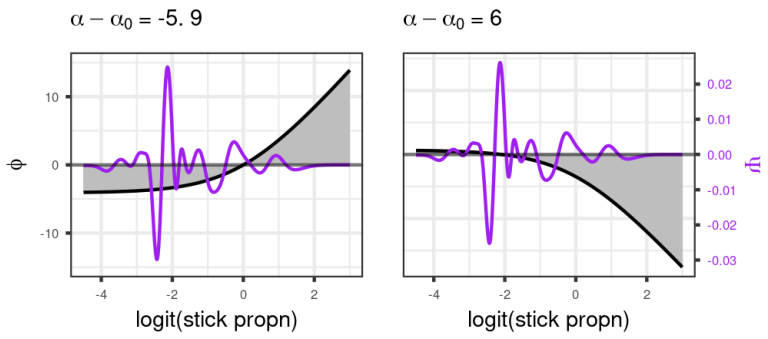


Figure 8: The multiplicative perturbations $\phi_\alpha(\cdot)$ that corresponds to decreasing (left) or increasing (right) the α parameter.

The population assignment z_{nli} is drawn from a multinomial distribution

$$p(z_{nli}|\pi_n) = \prod_{k=1}^{\infty} \pi_{nk}^{z_{nlik}}.$$

In this genetics application, we call π_n the *admixture* of individual n.

Initially we take $\mathcal{P}_{\text{stick}}$ to be $\text{Beta}(1, \alpha)$ with parameter $\alpha = \alpha_0 = 3$. The choice of $\alpha_0 = 3$ corresponds to roughly four distinct populations *a priori*, in agreement with the observation that the individuals come from four geographic regions. Below, we will evaluate sensitivity to this prior choice.

This model is identical to fastSTRUCTURE, a model proposed in Pritchard et al. [2000] and Raj et al. [2014], except that we replace the Dirichlet prior in fastSTRUCTURE with an infinite stick-breaking process. The result is a model similar to a hierarchical Dirichlet process for topic modeling [Teh et al., 2006], but without the top-level Dirichlet process. In addition, genotypes at genetic markers take the place of words in a document; in lieu of inferring "topics," we infer latent populations.

We use a mean-field variational approximation, and all distributions are conditionally conjugate except for the stick-breaking proportions, which remain logit-normal. See Appendix E.4 for further details.

Quantity of interest. The posterior quantities of interest in this application are the admixtures π_n. Figure 16 plots the inferred admixtures $\mathop{\mathbb{E}}_{\mathcal{Q}(\pi_n|\hat{\eta})}[\pi_n]$ for all individuals n.

In the approximate posterior with α_0, there appear to be three dominant latent populations, which we arbitrarily label as populations 1, 2, and 3 (top panel of Figure 9). The inferred admixture proportions generally correspond with geographic regions: Mbololo individuals are primarily population 1, Ngangao

individuals are primarily population 2, and Chawia individuals are a mixture of populations 1, 2, and 3 (Figure 16 in Appendix E.4).

Notably, outlying admixtures among individuals from the same geographic region provide clues into the historical migration patterns of this species. For example, while most Mbololo individuals are dominantly population 1, several Mbololo individuals have abnormally large admixture proportions of population 2. Conversely, while most Ngangao individuals are dominantly population 2, several Ngangao individuals have abnormally large admixture proportions of population 1. These patterns suggest that some migration has occurred between the Mbololo and Ngangao regions.

We evaluate the sensitivity of this conclusion to possible prior perturbations. Define the posterior quantity

$$g_{\text{admix}}(\eta; \mathcal{N}, k) = \mathop{\mathbb{E}}_{\mathcal{Q}(\pi|\eta)} \left[\frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \pi_{nk} \right],$$

the average admixture proportion of population k in a set of individuals $\mathcal{N}$.

Below, we consider g_{admix} with three different sets of individuals: $\mathcal{N} = \{26, ..., 31\}$, corresponding to the outlying Mbololo individuals, labeled "A" in Figure 9; $\mathcal{N} = \{125, ..., 128\}$, corresponding to the four outlying Ngangao individuals, labeled "B"; and $\mathcal{N} = \{139, ..., 155\}$ corresponding to all Chawia individuals, labeled "C". For individuals A, we let $k = 2$ in g_{admix} and examine the robustness of the presence of population 2; for individuals B, we use $k = 1$; and for individuals C, we use $k = 3$. The first two posterior quantities relate to the inferred migration between the Mbololo and Ngangao regions. In the last example, we study the robustness of having a third latent population present, a population that primarily appears in Chawia individuals.

Functional sensitivity. We construct worst-case negative perturbations for each of the three variants of g_{admix}, in order to see whether the biologically interesting patterns can be made to disappear with different prior choices. Figure 9 shows the result of the worst-case perturbations on the prior density and g_{admix}. After the worst-case perturbation, the admixture proportion of population 2 in individuals A was nearly halved. On the other hand, the admixture of population 1 in individuals B is more robust. We conclude that the inferred migration from Ngangao to Mbololo is relatively robust to the stick-breaking prior, while conclusions about migration from Mbololo to Ngangao may be dependent on prior choices.

In this data set and model, the conclusions from the linear approximation did not always agree with the conclusions from refitting the variational approximation. For example, the admixture proportion of population 3 in individuals C were predicted to more sensitive by our linear approximation than were actually observed after refitting (Figure 9, bottom row).

Moreover, even though the linear approximation agreed with the refits for individuals A in overall admixture proportion (Figure 9, second row), the approximation does not perform uniformly well over all individuals. Figure 10 plots the inferred admixtures computed using the linearized variational parameters and

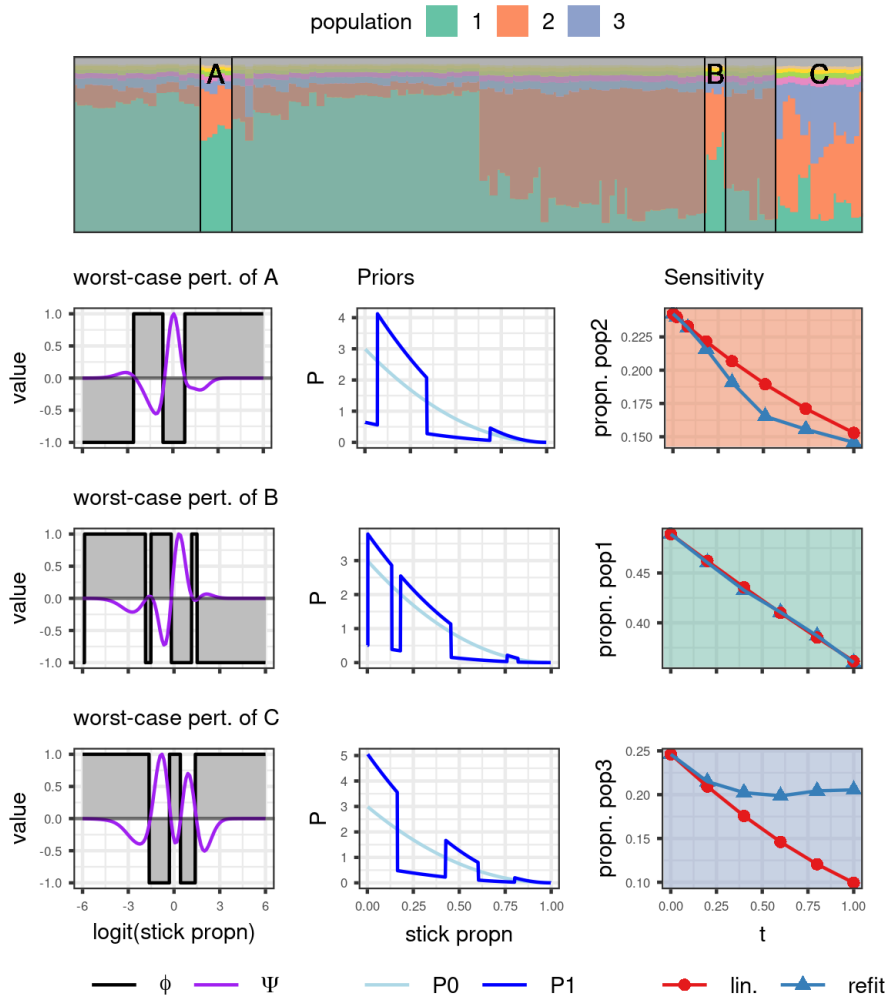


Figure 9: Sensitivity of inferred admixtures for several outlying individuals. For individuals A, we examine the sensitivity of the admixture proportion of population 2. For individuals B, we examine the population 1 admixture. For the individuals C, we examine the population 3 admixture. (Left column) The worst-case negative perturbation with $\|\phi\|_\infty = 1$ in grey, plotted against the influence function in purple (scaled such that $\|\psi\|_\infty = 1$). (Middle column) The effect of the perturbation on the prior density. (Right column) Effects on the inferred admixture.

the refitted variational parameters. The admixture proportion of population 2 in individual $n = 25$ dramatically increased after refitting with the perturbed prior;

the linearized parameters failed to reproduce this change.

Even though linear approximation works less well in this example, the influence function is still able to guide our choice of functional perturbations at which to refit. While the worst-case perturbations we used may be an adversarial choice, the influence function suggests that we can construct a smoother perturbation with a similar effect as the worst-case, as we did in Section 7.2. Importantly, as we will note in the next subsection, the influence function is cheap to compute relative to refitting. For a further discussion of the limitations of the linear approximation, see Appendix F.

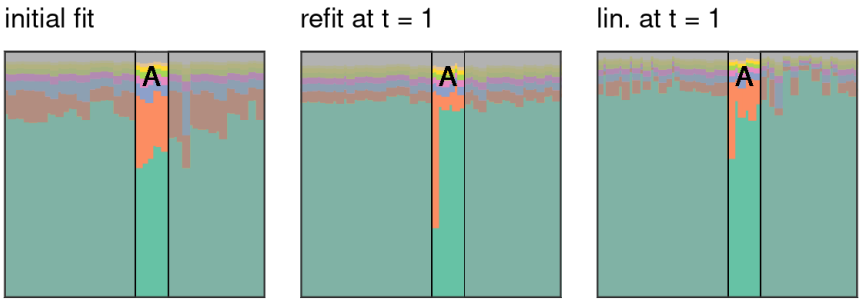


Figure 10: Inferred admixtures after the worst-case perturbation to individuals "A" (see Figure 9 for perturbation).

7.4 Computation time

The relative computational costs of the approximation and re-fitting for our three experiments are shown in Table 1. The data sets we considered in our experiments had varying degrees of complexity, and the computational of cost of fitting the variational approximation thus also varies accordingly. However, the cost of forming the linear approximation—the step that requires computing and inverting the Hessian matrix—was consistently roughly an order of magnitude faster than refitting.

Recall from Section 6 that the solution of a linear system involving $\hat{H}^{-1}$ is the computationally intensive part of the linear approximation, and that the linear system needs to be solved only once for a given perturbation, as described in Section 6. Consistent with this observation, in all the examples, after the linear approximation is formed, extrapolating to any new prior parameter $\alpha \neq \alpha_0$ or $t \neq 0$ takes only fractions of a second. For example, in the thrush data and fastSTRUCTURE model, the initial fit took seven seconds, with subsequent refits (which we warm-started with the initial fit) taking between five and ten seconds. Solving a linear system to form the linear approximation for a particular

Table 1: Compute time in seconds of various quantities on each data set. Reported times for $\hat{\eta}(\alpha)$ and $\hat{\eta}^{\text{lin}}(\alpha)$ are median times over the set of considered α's. The reported influence function time is the time required to evaluate the influence function on a grid of 1000 points.

	iris	mice	thrush
Initial fit	1	30	7
Hessian solve for α sensitivity	0.02	3	0.3
Linear approx. $\hat{\eta}^{\text{lin}}(\alpha)$	0.0008	0.001	0.0008
Refits $\hat{\eta}(\alpha)$	0.5	30	5
The influence function (at 1000 grid points)	0.09	3	0.6
Hessian solve for ϕ	0.02	3	0.4
Linear approx. $\hat{\eta}^{\text{lin}}(\phi)$	0.001	0.001	0.0008
Refit $\hat{\eta}^{\text{lin}}(\phi)$	0.6	20	10

perturbation ϕ took less than a second, and evaluating $\hat{\eta}(\phi)$ was essentially free.

Acknowledgments. We are indebted to helpful discussions with Nelle Varoquaux, Matthew Stephens, Michael C. Hughes, Eric Sudderth, and Jake Soloff. Runjing Liu is supported by the National Science Foundation graduate research fellowship program. Ryan Giordano and Tamara Broderick were supported in part by an NSF CAREER Award and an ONR Early Career Grant.

References

E. Anderson. The species problem in iris. *Annals of the Missouri Botanical Garden*, 23(3):457–509, 1936. 17

V. Averbukh and O. Smolyanov. The theory of differentiation in linear topological spaces. *Russian Mathematical Surveys*, 22(6):201–258, 1967. 46

Sanjib Basu. *Bayesian Robustness and Bayesian Nonparametrics*, pages 223–240. Springer New York, New York, NY, 2000. 14

A. Baydin, B. Pearlmutter, A. Radul, and J. Siskind. Automatic differentiation in machine learning: A survey. *Journal of Machine Learning Research*, 18, 2018. 9

P. Billingsley. *Probability and Measure*. John Wiley and Sons, second edition, 1986. 35, 43

C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. 53

D. Blackwell and J. B. MacQueen. Ferguson distributions via Polya urn schemes. 57

D. M. Blei and M. I. Jordan. Variational inference for Dirichlet process mixtures. *Bayesian Analysis*, 1(1):121 – 143, 2006. 5

D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017. 5, 53

D. Cai, T. Campbell, and T. Broderick. Power posteriors do not reliably learn the number of components in a finite mixture. In *ICBINB@NeurIPS workshop*, 2020. 14

D. Cai, T. Campbell, and T. Broderick. Finite mixture models do not reliably learn the number of components. In *Proceedings of the 38th International Conference on Machine Learning (to appear)*, 2021. 14

D. Cook. Assessment of local influence. *Journal of the Royal Statistical Society: Series B (Methodological)*, 48(2):133–155, 1986. 14

R. Dudley. *Real Analysis and Probability*. CRC Press, 2018. 11, 41

T. S. Ferguson. A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, pages 209–230, 1973. 1

P. Galbusera, L. Lens, T. Schenck, E. Waiyaki, and E. Matthysen. Genetic variability and gene flow in the globally, critically-endangered taita thrush. *Conservation Genetics*, 1:45–55, 2000. 22

Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. *Bayesian Data Analysis, Third Edition*. Chapman & Hall/CRC Texts in Statistical Science. Taylor & Francis, 2013. 2, 14

R. Giordano, T. Broderick, and M. I. Jordan. Covariances, robustness and variational Bayes. *Journal of Machine Learning Research*, 19(51), 2018. 3, 14

P. Gustafson. Local sensitivity of posterior expectations. *Annals of Statistics*, 24 (1):174–195, 1996a. 3, 7, 8, 10, 12, 13, 14, 39, 41, 42

P. Gustafson. Local sensitivity of inferences to prior marginals. *Journal of the American Statistical Association*, 91(434):774–781, 1996b. 14

P. Gustafson. *Local Robustness in Bayesian Analysis*, pages 71–88. Springer New York, New York, NY, 2000. 14

F. Hampel, E. Ronchetti, P. Rousseeuw, and W. Stahel. *Robust Statistics: The Approach Based on Influence Functions*, volume 196. John Wiley & Sons, 2011. 14

D. R. Insua and F Ruggeri. *Robust Bayesian Analysis*. Springer, 2000. 14

L. Jaeckel. The infinitesimal jackknife, memorandum. Technical report, MM 72-1215-11, Bell Lab. Murray Hill, NJ, 1972. 14

A. Jasra, C. C. Holmes, and D. A. Stephens. Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modeling. *Statistical Science*, 20(1):50 – 67, 2005. 3

S. Krantz and H. Parks. *The Implicit Function Theorem: History, Theory, and Applications*. Springer Science & Business Media, 2012. 34, 45

A. Kucukelbir, D. Tran, R. Ranganath, A. Gelman, and D. Blei. Automatic differentiation variational inference. *arXiv preprint arXiv:1603.00788*, 2016. 3, 37

Y. Luan and H. Li. Clustering of time-course gene expression data using a mixed-effects model with B-splines. *Bioinformatics*, 19(4):474–482, 2003. 20, 58

J. W. Miller and M. T. Harrison. A simple example of dirichlet process mixture inconsistency for the number of components. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. 14

J. W. Miller and M T. Harrison. Inconsistency of Pitman-Yor process mixtures for the number of components. *Journal of Machine Learning Research*, 15(96): 3333–3370, 2014. URL http://jmlr.org/papers/v15/miller14a.html. 14

O. Nielsen. *An Introduction to Integration and Measure Theory*, volume 17. Wiley-Interscience, 1997. 43, 49

Luis E. Nieto-Barajas and Igor Prünster. A sensitivity analysis for Bayesian nonparametric density estimators. *Statistica Sinica*, 19(2):685–705, 2009. 14

J. Nocedal and S. Wright. *Numerical Optimization*. Springer Science & Business Media, 2006. 15, 16

J. K. Pritchard, M. Stephens, and P. Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155(2):945–959, 2000. 25

A. Raj, M. Stephens, and J. K. Pritchard. fastSTRUCTURE: Variational inference of population structure in large SNP data sets. *Genetics*, 197(2):573–589, 2014. 25

Rajesh Ranganath, Sean Gerrish, and David M. Blei. Black box variational inference, 2013. https://arxiv.org/abs/1401.0118. 3

J. Reeds. *On the definition of von Mises functionals*. PhD thesis, Statistics, Harvard University, 1976. 46

A. Saha and S. Kurtek. Geometric sensitivity measures for Bayesian nonparametric density estimation models. *Sangkhya Series A.*, 81:104–143, 2019. 14

J. Sethuraman. A constructive definition of Dirichlet priors. *Statistica sinica*, pages 639–650, 1994. 1

J. E. Shoemaker, S. Fukuyama, A. J. Eisfeld, et al. An ultrasensitive mechanism regulates influenza virus-induced inflammation. *PLoS Pathogens*, 11(6):1–25, 2015. 20, 57

S. Sivaganesan. Global and local robustness approaches: Uses and limitations. In *Robust Bayesian Analysis*, pages 89–108. Springer, 2000. 14

J. D. Storey, W. Xiao, J. T. Leek, R. G. Tompkins, and R. W. Davis. Significance analysis of time course microarray experiments. *Proceedings of the National Academy of Sciences of the United States of America*, 102(36):12837–42, 2005. 57

Y. W. Teh. Dirichlet processes. In *Encyclopedia of Machine Learning*. Springer, 2010. 57

Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006. 2, 25

P. Virtanen, R. Gommers, T. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. van der Walt, M. Brett, J. Wilson, J. Millman, N. Mayorov, A. Nelson, E. Jones, R. Kern, E. Larson, C. Carey, İ Polat, Y. Feng, E. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. Quintero, C. Harris, A. Archibald, A. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17: 261–272, 2020. doi: 10.1038/s41592-019-0686-2. 15

U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17: 395–416, 2007. 22

E. Zeidler. *Nonlinear Functional Analysis and Its Applications I: Fixed point theorems*. Springer Verlag New York, Inc., 1986. 11, 12, 13, 46, 48, 49

Appendices

A General differentiability results

Our goal is to approximate the dependence of the optimal VB parameters on the prior using a Taylor series, which requires that the optimal VB parameters must be continuously differentiable as a function of the prior specification. In this section we state general conditions under which VB optima based on reverse KL

divergence are differentiable functions of both parametric and nonparametric prior perturbations. We will state our conditions and results in terms of a generic VB approximation and prior perturbation, which we articulate in Definition 2.

Definition 2. For some parameter $\theta \in \Omega_\theta \subseteq \mathbb{R}^{D_\theta}$, let $\mathcal{P}(\theta|t)$ denote a class of probability densities relative to a sigma-finite measure μ, defined for t in an open set $\mathcal{B}_t \subseteq \mathbb{R}$ containing 0. Let $\mathcal{Q}(\theta|\eta)$ be a family of approximating densities, also defined relative to μ.

Let the variational objective factorize as

$$\mathrm{KL}(\eta, t) := \mathrm{KL}(\eta) - \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[(\log \mathcal{P}(\theta|t) - \log \mathcal{P}(\theta|t=0))\right] \tag{22}$$

$$\hat{\eta}(t) := \mathop{\mathrm{argmin}}_{\eta \in \Omega_\eta} \mathrm{KL}(\eta, t). \tag{23}$$

Let $\hat{\eta}$ with no argument refer to $\hat{\eta}(0)$, the minimizer of $\mathrm{KL}(\eta)$. ⊡

In general, we expect θ of Definition 2 to be some subset of the model parameters whose prior is being perturbed. The decomposition in Eq. 22 is always possible for VB approximations based on reverse KL divergence, in the sense that one could always take θ to be all model parameters and $\mathrm{KL}(\eta) = 0$. We decompose the objective in this way in order to state strict regularity assumptions only on the part of the reverse KL divergence that is being perturbed. Indeed, we will require little from the $\mathrm{KL}(\eta)$ part of the decomposition other than that it can be differentiated and optimized.

By identifying t with some hyperparameter (e.g. the concentration parameter, as in Example 1 below), we can use Definition 2 to study parametric perturbations. Furthermore, by parameterizing a path through the space of general densities, Definition 2 will allow us to study nonparametric perturbations (e.g. Example 2 below and the detailed analysis of Appendices A.3 to A.5). We can thus study VB prior robustness in general by studying problems of the form in Definition 2.

Example 1. For the BNP model with the $\mathcal{P}_{\text{stick}}$ prior on the stick breaks, take $\theta = (\nu_1, \ldots, \nu_{K_{\max}-1})$, and take μ to be the Lebesgue measure on $[0,1]^{K_{\max}-1}$. Let α_0 be some initial value of the concentration parameter, and let t be $\alpha - \alpha_0$, so that deviations of t away from 0 represent deviations of α away from α_0.

Expanding the reverse KL divergence in Eq. 3, we see that the prior $\mathcal{P}(\nu_k|\alpha)$ enters the VB objective in a term of the form $\sum_{k=1}^{\infty} \mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta)}[\log \mathcal{P}(\nu_k|\alpha)]$. Adding and subtracting the this term evaluated at α_0 gives

$$\mathrm{KL}(\eta, \alpha) = \mathrm{KL}(\eta, \alpha_0) - \sum_{k=1}^{K_{\max}-1} \left(\mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta)}[\log \mathcal{P}(\nu_k|\alpha)] - \mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta)}[\log \mathcal{P}(\nu_k|\alpha_0)] \right).$$

Plugging in the definition of $\mathcal{P}(\nu_k|\alpha)$, recognizing that the normalizing constant does not depend on ν_k and so can be neglected in the optimization, letting

$\mathrm{KL}(\eta) := \mathrm{KL}(\eta, \alpha_0)$, and substituting $t = \alpha - \alpha_0$ gives

$$\mathrm{KL}(\eta, t) = \mathrm{KL}(\eta, \alpha_0) - t \sum_{k=1}^{K_{\max}-1} \mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta)} [\log(1-\nu_k)].$$

△

Example 2. As in Example 1, take $\theta = (\nu_1, \ldots, \nu_{K_{\max}-1})$ and μ to be the Lebesgue measure on $[0,1]^{K_{\max}-1}$. Let $\mathcal{P}_0(\nu_k) := \mathrm{Beta}(\nu_k|1, \alpha_0)$, and let $\mathcal{P}_1(\nu_k)$ be a density, not in the beta family, that shifts mass towards zero:

$$\mathcal{P}_1(\nu_k) := \frac{\exp(-\nu_k)\mathcal{P}_0(\nu_k)}{\int \exp(-\nu'_k)\mathcal{P}_0(\nu'_k)d\nu'_k}.$$

For $t \in [0,1]$ define the multiplicatively perturbed prior

$$\mathcal{P}(\nu_k|t) := \frac{\mathcal{P}_1(\nu_k)^t \mathcal{P}_0(\nu_k)^{1-t}}{\int \mathcal{P}_1(\nu'_k)^t \mathcal{P}_0(\nu'_k)^{1-t} d\nu'_k}.$$

When $t = 0$, $\mathcal{P}(\nu_k|t) = \mathcal{P}_0(\nu_k)$, when $t = 1$, $\mathcal{P}(\nu_k|t) = \mathcal{P}_1(\nu_k)$. For $t \in (0,1)$ $\mathcal{P}(\nu_k|t)$ varies smoothly between $\mathcal{P}_0$ and $\mathcal{P}_1$.

As in Example 1, up to constants not depending on ν_k we can write

$$\begin{aligned}
\log \mathcal{P}(\nu_k|t) - \log \mathcal{P}(\nu_k|t=0) &= -t \log \mathcal{P}_0(\nu_k) + t \log \mathcal{P}_1(\nu_k) + C \\
&= -t\nu_k + C \Rightarrow \\
\mathrm{KL}(\eta, t) &= \mathrm{KL}(\eta) - t \mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta)} [\nu_k] + C.
\end{aligned}$$

Different choices for $\mathcal{P}_1(\nu_k)$ would give different additive perturbations to the reverse KL divergence. △

A.1 Parametric prior perturbations

We now state conditions under which $t \mapsto \hat{\eta}(t)$, as defined by Definition 2, is continuously differentiable. Our key theoretical tool will be the implicit function theorem [see, e.g., Krantz and Parks, 2012], applied to the first-order conditions for the VB optimization problem.

Our results can be expressed in terms of unnormalized densities, which can simplify some computation. To that end, let $\tilde{\mathcal{Q}}$ and $\tilde{\mathcal{P}}$ refer to potentially unnormalized (but normalizable) versions of the respectively corresponding $\mathcal{Q}$ and $\mathcal{P}$ given in Definition 2, so that

$$\mathcal{Q}(\theta|\eta) := \frac{\tilde{\mathcal{Q}}(\theta|\eta)}{\int \tilde{\mathcal{Q}}(\theta'|\eta)\mu(d\theta')} \quad \text{and} \quad \mathcal{P}(\theta|t) := \frac{\tilde{\mathcal{P}}(\theta|t)}{\int \tilde{\mathcal{P}}(\theta'|t)\mu(d\theta')}.$$

In Assumption 1, stated in Section 3 above, we require some mild regularity conditions for the "initial problem," $\mathrm{KL}(\eta)$. As we discuss in Appendix A.2,

Assumption 1 states conditions that are typically satisfied when $\mathrm{KL}(\eta)$ can be optimized numerically using unconstrained optimization.

Next, we will require some differentiability conditions for the perturbation and the variational approximation.

Assumption 2. *Assume that the map $\eta \mapsto \log \tilde{\mathcal{Q}}(\theta|\eta)$ is twice continuously differentiable, and that the map $t \mapsto \log \tilde{\mathcal{P}}(\theta|t)$ is continuously differentiable.*

Further, assume that we can exchange the order of integration and differentiation in the expressions $\int \tilde{\mathcal{Q}}(\theta|\eta) \log \tilde{\mathcal{P}}(\theta|t)\mu(d\theta)$ and $\int \tilde{\mathcal{Q}}(\theta|\eta)\mu(d\theta)$ at $\eta = \hat{\eta}$ and $t = 0$ for the derivatives $\partial/\partial\eta$, $\partial^2/\partial\eta^2$, and $\partial^2/\partial\eta\partial t$.

In certain cases, one can verify Assumption 2 directly, such as when $\mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\log \tilde{\mathcal{P}}(\theta|t)\right]$ has a closed form. For more general situations, the following assumptions allow us to satisfy Assumption 2 using the dominated convergence theorem [Billingsley, 1986, Theorem 16.8].

Assumption 3. *Let $f(\theta, \eta, t)$ be a function taking values in $\mathbb{R}$. Assume that the partial derivatives $\partial/\partial\eta$, $\partial^2/\partial\eta^2$, and $\partial^2/\partial\eta\partial t$ of f exist, are continuous functions of η and t, and are μ-measureable functions of θ on some open set $\mathcal{B}_\eta \times \mathcal{B}_t$.*

Let $M(\theta) > 0$ be a measurable function with $\int M(\theta)\mu(d\theta) < \infty$. Assume that, for all $\eta, t \in \mathcal{B}_\eta \times \mathcal{B}_t$, $M(\theta)$ is μ-almost everywhere greater than each of the following functions: $|f(\theta, \eta, t)|$, $\|\partial f(\theta, \eta, t)/\partial\eta\|_2$, $\left\|\partial^2 f(\theta, \eta, t)/\partial\eta\partial\eta^T\right\|_2$, and $\left\|\partial^2 f(\theta, \eta, t)/\partial\eta\partial t\right\|_2$.

Assumption 4. *(Sufficient conditions for Assumption 2.) Let Assumption 3 hold with the function $f(\theta, \eta, t) = \tilde{\mathcal{Q}}(\theta|\eta) \log \tilde{\mathcal{P}}(\theta|t)$ as well as with $f(\theta, \eta, t) = \tilde{\mathcal{Q}}(\theta|\eta)$.*

By the dominated convergence theorem, Assumption 4 implies Assumption 2 (see Lemma 2 in Appendix B for a proof). The advantage of Assumption 4 over Assumption 2 is that the conditions of Assumption 4 can typically be verified even when the expectation $\mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\log \tilde{\mathcal{P}}(\theta|t)\right]$ does not have a closed form. In Appendix A.2, we will discuss how different choices of variational approximations for the stick lengths lend themselves to either Assumption 2 of Assumption 4. Furthermore, Assumption 3 will be essential to analyzing nonparametric perturbations in Appendix A.3.

We are now in a position to define the quantities that occur in the derivative and state our main result.

Definition 3. Under the conditions of Definition 2, when Assumptions 1 and 2 hold, define

$$\hat{H} := \left.\frac{\partial^2 \mathrm{KL}(\eta)}{\partial\eta\partial\eta^T}\right|_{\hat{\eta}} \quad \text{and}$$

$$\mathcal{S}(\theta|\eta) := \nabla_\eta \log \tilde{\mathcal{Q}}(\theta|\eta) - \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\nabla_\eta \log \tilde{\mathcal{Q}}(\theta|\eta)\right].$$

Further, define

$$\hat{J} := \left.\frac{\partial \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\left.\frac{\partial \log \tilde{\mathcal{P}}(\theta|t)}{\partial t}\right|_{t=0}\right]}{\partial \eta}\right|_{\eta=\hat{\eta}} = \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\hat{\eta})}\left[\mathcal{S}(\theta|\hat{\eta})\left.\frac{\partial \log \tilde{\mathcal{P}}(\theta|t)}{\partial t}\right|_{t=0}\right],$$

where the final equality follows from differentiating under the integral using Assumption 2 (see Lemma 3 in Appendix B for more details). ⊡

Theorem 4. *Under the conditions of Definitions 2 and 3, let Assumptions 1 and 2 hold. Then the map $t \mapsto \hat{\eta}(t)$ is continuously differentiable at $t = 0$ with derivative*

$$\left.\frac{d\hat{\eta}(t)}{dt}\right|_0 = -\hat{H}^{-1}\hat{J}. \tag{24}$$

(For a proof, see Appendix B Proof B.)

A.2 Differentiability of BNP models with respect to α

In this section, we return to the BNP problem and prove carefully that the map $\alpha \mapsto \hat{\eta}(\alpha)$ satisfies Assumptions 1 and 2, and so the conditions of Theorem 4. As in Example 1, we will take μ to be the Lebesgue measure on $[0,1]^{K_{\max}-1}$.

Recall from Section 2.2 that we take $\mathcal{Q}(\nu_k|\eta)$ to be a normal density on the logit-transformed sticks, $\tilde{\nu}_k$. For the duration of this section, write $\mathcal{Q}(\tilde{\nu}_k|\eta) = \mathcal{N}\left(\tilde{\nu}_k|\mu_k, \sigma_k^2\right)$, so that the subvector of η parameterizing $\mathcal{Q}(\tilde{\nu}_k|\eta)$ is $\eta_{\nu_k} = (\mu_k, \sigma_k)$. By the formula for transformation of probability densities,

$$\mathcal{Q}(\nu_k|\eta_{\nu_k}) = \mathcal{N}\left(\log\left(\frac{\nu_k}{1-\nu_k}\right)\middle|\mu_k, \sigma_k^2\right)\frac{1}{\nu_k(1-\nu_k)},$$

where we have used the fact that $\left.\frac{d\tilde{\nu}_k}{d\nu_k}\right|_{\nu_k} = \frac{1}{\nu_k(1-\nu_k)}$. Similarly, for any function $f(\nu_k)$ of the stick lengths, we can transform the expectations as $\mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta_{\nu_k})}[f(\nu_k)] = \mathop{\mathbb{E}}_{\mathcal{Q}(\tilde{\nu}_k|\eta_{\nu_k})}\left[f\left(\frac{\exp(\tilde{\nu}_k)}{1+\exp(\tilde{\nu}_k)}\right)\right]$, using the fact that $\nu_k = \frac{\exp(\tilde{\nu}_k)}{1+\exp(\tilde{\nu}_k)}$.

Differentiability of $\mathrm{KL}(\eta)$ (Assumption 1 (Item (1))) is immediately satisfied for the η that parameterize $\mathcal{Q}(\beta|\eta)$ and $\mathcal{Q}(z|\eta)$ by our use of conjugate approximating families and standard parameterizations. The stick length density, $\mathcal{Q}(\nu_k|\eta_{\nu_k})$ is not a standard exponential family [2] , so we must show that the entropy

[2] In this section, we continue to take μ to be the Lebesgue measure on $[0,1]$ as in Example 1. We could have equivalently taken μ to be the Lebesgue measure on $\mathbb{R}$ and analyzed $\mathcal{P}(\tilde{\nu}_k|\alpha)$ instead of $\mathcal{P}(\nu_k|\alpha)$. Had we done so, the log Jacobian term $\log(\nu_k(1-\nu_k))$ now appearing in the entropy would have instead appeared in the $\log\tilde{\mathcal{P}}(\tilde{\nu}_k|\alpha)$ term, and so been part of Assumption 2 rather than Assumption 1 (Item (1)). Nevertheless, the needed assumptions would be substantively the same. For essentially this reason, the choice of dominating measure in Definition 2 does not matter.

$\mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta_{\nu_k})}[\log \mathcal{Q}(\nu_k|\eta_{\nu_k})]$ is twice continuously differentiable. The entropy is given up to a constant by

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta_{\nu_k})}[\log \mathcal{Q}(\nu_k|\eta_{\nu_k})] \\
&= \mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta_{\nu_k})}\left[\log \mathcal{N}\left(\log\left(\frac{\nu_k}{1-\nu_k}\right)\middle|\mu_k, \sigma_k^2\right)\right] + \mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta_{\nu_k})}[\log(\nu_k(1-\nu_k))] \\
&= \mathop{\mathbb{E}}_{\mathcal{Q}(\tilde{\nu}_k|\eta_{\nu_k})}\left[\log \mathcal{N}\left(\tilde{\nu}_k\middle|\mu_k, \sigma_k^2\right)\right] + \mathop{\mathbb{E}}_{\mathcal{Q}(\tilde{\nu}_k|\eta_{\nu_k})}[\tilde{\nu}_k] \\
&= \frac{1}{2}\log\sigma_k^2 + \mu_k + C,
\end{aligned}$$

which is twice continuously differentiable by inspection. Indeed, Assumption 1 (Item (1)) is typically satisfied in VB problems; when it is not, many black-box optimization methods also do not apply.

Non-singularity of the Hessian matrix $\hat{H}$ (Assumption 1 (Item (2))) is satisfied whenever $\hat{\eta}$ is at a local optimum of $\mathrm{KL}(\eta)$. In practice, we compute $\hat{\eta}$ and (approximately) check Assumption 1 (Item (2)) numerically as part of computing the sensitivity $\hat{H}^{-1}\hat{J}$. As with Assumption 1 (Item (1)), if Assumption 1 (Item (2)) is violated, then the user will probably have difficulty optimizing $\mathrm{KL}(\eta)$.

Assumption 1 (Item (3)) essentially requires that $\mathrm{KL}(\eta)$ be well-defined in an $\mathbb{R}^{D_\eta}$ neighborhood of $\hat{\eta}$, and can require some care in choosing the parameterization η. As an example of a parameterization that would violate Assumption 1 (Item (3)), consider parametrizing $\mathcal{Q}(z_n|\eta)$ by the $K_{\max}$ expectations $m_k := \mathop{\mathbb{E}}_{\mathcal{Q}(z_n|\eta)}[z_{nk}]$. The set $(m_1, \ldots, m_{K_{\max}})$ completely specify $\mathcal{Q}(z_n|\eta)$, but violate Assumption 1 (Item (3)), since any valid parameterization satisfies $\sum_{k=1}^{K_{\max}} m_k = 1$, and so no open ball in $\mathbb{R}^{D_\eta}$ can be contained in Ω_η. However, Assumption 1 (Item (3)) is satisfied we use an *unconstrained parameterization* for $\mathcal{Q}(\zeta|\eta)$. Unconstrained parameterizations of variational distributions allow the use of unconstrained optimization for variational inference and are a good practice when available [Kucukelbir et al., 2016]. For details on our parameterizations, see the corresponding appendices.

Verifying Assumption 2 is the principal technical challenge of satisfying the conditions of Theorem 4. Recall from Example 1 that $\log\tilde{\mathcal{P}}(\nu_k|t) = t\log(1-\nu_k)$, so we need to establish Assumption 2 for

$$-\mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta_{\nu_k})}[t\log(1-\nu_k)] = \mathop{\mathbb{E}}_{\mathcal{Q}(\tilde{\nu}_k|\eta_{\nu_k})}[t\log(1+\exp(\tilde{\nu}_k))]\,.$$

Since the preceding equality holds for all t and η_{ν_k}, it suffices to establish that we can exchange the order of integration and differentiation for the right hand side. Since the normal density has a term of the form $\exp(-C\tilde{\nu}_k^2)$, and since $\log(1+\exp(\tilde{\nu}_k))\exp(-|\tilde{\nu}_k|) < \infty$ for all $\tilde{\nu}_k \in \mathbb{R}$ as long as the variational variance is finite, one can show that the conditions of Assumption 4 are satisfied within $\mathcal{B}_\eta \times \mathcal{B}_t$. (See Lemma 7 in Appendix B for a proof.) Note that derivatives with

respect to any components of η other than η_{ν_k} are zero and so Assumption 2 is trivially satisfied.

Assumption 4 implies Assumption 2. Since both Assumptions 1 and 2 are satisfied, Theorem 4 applies, and the map $\alpha \mapsto \hat{\eta}(\alpha)$ is continuously differentiable.

We end this section by observing that the only real technical challenge was showing that the assumptions were satisfied for the logit-normal densities $\mathcal{Q}(\nu_k|\eta_{\nu_k})$. Had we instead used the conjugate beta density parameterized by its natural parameters, then both Assumption 1 and Assumption 2 would follow immediately by standard properties of the Beta distribution. In particular, the expectation $\mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta_{\nu_k})}[t\log(1-\nu_k)]$ needed for Assumption 1 is simply t times the Beta distribution's moment parameter, which is known to be an infinitely-differentiable function of the natural parameters.

A.3 Nonparametric prior perturbations

We now show how, by parameterizing a path between two arbitrary densities, we can apply Theorem 4 to nonparametric perturbations of the prior density. Again let us return to the abstract setting of Definition 2. Let us fix an initial prior density, $\mathcal{P}_0(\theta)$, at which we have computed a VB approximation, and suppose we wish to ask what the variational optimum would have been had we used some alternative prior density, $\mathcal{P}_1(\theta)$. Let us write $\hat{\eta}(\mathcal{P}_0)$ and $\hat{\eta}(\mathcal{P}_1)$ for these two approximations, respectively, so we are interested in quantifying the change $g(\hat{\eta}(\mathcal{P}_1)) - g(\hat{\eta}(\mathcal{P}_0))$. If this change is large, we say that our quantity of interest is not robust to replacing $\mathcal{P}_0$ with $\mathcal{P}_1$.

To approximately assess robustness using the local sensitivity approach, we must somehow define a continuous path from $\mathcal{P}_0(\theta)$ to $\mathcal{P}_1(\theta)$ parameterized, say, by $t \in [0,1]$. One way to do so is to define a multiplicative path

$$\log \tilde{\mathcal{P}}(\theta|t) = (1-t)\log \mathcal{P}_0(\theta) + t \log \mathcal{P}_1(\theta). \tag{25}$$

Under Eq. 25, when $t=0$, $\mathcal{P}(\theta|t) = \mathcal{P}_0(\theta)$, when $t=1$, $\mathcal{P}(\theta|t,\mathcal{P}_0,\mathcal{P}_1) = \mathcal{P}_1(\theta)$, and $t \in (0,1)$ smoothly parameterizes a path between the two. If we can verify that Theorem 4 applies to the perturbation given in Eq. 25, then, just as in the parametric case, we can form the Taylor series approximation,

$$\hat{\eta}(\mathcal{P}_1) \approx \hat{\eta}(\mathcal{P}_0) + \left.\frac{d\hat{\eta}(t)}{dt}\right|_{t=0} (1-0).$$

Our first task is then to state conditions under which Theorem 4 applies to Eq. 25. In Eq. 25 we have assumed that $\mathcal{P}_1$ is a density, but it will be more convenient to observe that, when $\mathcal{P}_1 \ll \mathcal{P}_0$, we can re-write

$$\log \tilde{\mathcal{P}}(\theta|t) = \log \mathcal{P}_0(\theta) + t \log \frac{\tilde{\mathcal{P}}_1(\theta)}{\tilde{\mathcal{P}}_0(\theta)} + C. \qquad (C \text{ does not depend on } \theta)$$

Defining the generic function $\phi(\theta) := \log \frac{\tilde{\mathcal{P}}_1(\theta)}{\tilde{\mathcal{P}}_0(\theta)}$ motivates consideration of perturbations of the form $\log \tilde{\mathcal{P}}(\theta|t) = \mathcal{P}_0(\theta) + t\phi(\theta)$, where $\phi(\theta)$ is some generic measurable function. We can then ask what ϕ give rise to valid densities as well as differentiable maps $t \mapsto \hat{\eta}(t)$.

Definition 4. Let μ denote a measure on the Borel sets of Ω_θ and fix $\mathcal{P}_0(\theta)$, a density with respect to μ. Assume that $\mu(\{\theta : \mathcal{P}_0(\theta) = 0\}) = 0$, so that (in a slight abuse of notation) $\mu \ll \mathcal{P}_0$. For any measurable $\phi : \Omega_\theta \mapsto \mathbb{R}$ for which the expressions are well-defined, let

$$\tilde{\mathcal{P}}(\theta|\phi) := \mathcal{P}_0(\theta) \exp(\phi(\theta)).$$

As usual, when $0 < \int \tilde{\mathcal{P}}(\theta|\phi)\mu(d\theta) < \infty$, we let $\mathcal{P}(\theta|\phi)$ be the normalized version of $\tilde{\mathcal{P}}(\theta|\phi)$. Further, define the norm $\|\phi\|_\infty := \text{esssup}_{\theta\sim\mu} |\phi(\theta)|$, and let $\mathcal{B}_\phi(\delta) := \{\phi : \|\phi\|_\infty < \delta\}$. ⊡

The class of perturbations defined in Definition 4 are one of the family of "nonlinear" functional perturbations given by Gustafson [1996a], though we deviate from Gustafson [1996a] by allowing ϕ to take on negative values. The following result, which motivates the use of the $\|\cdot\|_\infty$ norm to measure the "size" of a perturbation ϕ, is only a minor modification of the corresponding result from Gustafson [1996a] to allow negative perturbations.

Lemma 1. *(Gustafson [1996a]) Fix the quantities given in Definition 4. For a fixed probability measure $\mathcal{P}_1 \ll \mu$ with density $\mathcal{P}_1(\theta)$ with respect to μ, let $\phi(\theta|\mathcal{P}_1) := \log \mathcal{P}_1(\theta)/\mathcal{P}_0(\theta)$. Then $\mathcal{P}_1 \mapsto \|\phi(\cdot|\mathcal{P}_1)\|_\infty$ is a norm, does not depend on μ, and is invariant to invertible transformations of θ.*

Furthermore, for any ϕ with $\|\phi\|_\infty < \infty$, the quantity $\tilde{\mathcal{P}}(\theta|\phi)$ gives rise to a valid prior, in the sense that $\tilde{\mathcal{P}}(\theta|\phi) \geq 0$ μ-almost everywhere, and $0 < \int \tilde{\mathcal{P}}(\theta|\phi)\mu(d\theta) < \infty$. (See Proof B on page 45.)

The set of priors $\{\mathcal{P}(\theta|\phi) : \phi \in \mathcal{B}_\phi(\delta)\}$ live in a multiplicative band around the original prior, $\mathcal{P}_0$, as shown in Figure 1. Although Lemma 1 proves that every ϕ with $\|\phi\|_\infty$ is a valid prior, the converse is not true, and the Beta prior perturbation of Example 1 is a counterexample.

Example 3. Take μ to be the Lebesgue measure on $[0, 1]$, let $\mathcal{P}_0(\theta) = \text{Beta}(\theta|1, \alpha_0)$ and $\mathcal{P}_1(\theta) = \text{Beta}(\theta|1, \alpha_1)$ for $\alpha_0 \neq \alpha_1$. Taking $\phi(\theta) = (\alpha_1 - \alpha_0)\log(1 - \theta)$ parameterizes a path from $\mathcal{P}_0$ to $\mathcal{P}_1$ as in Eq. 25, and

$$\|\phi\|_\infty = |\alpha_1 - \alpha_0| \sup_{\theta\in[0,1]} |\log(1 - \theta)| = \infty.$$

Therefore, in general, there exist valid priors that cannot be expressed by Definition 4 with ϕ with $\|\phi\|_\infty < \infty$. △

We now show that, when $\|\phi\|_\infty < \infty$, we can apply Theorem 4. We still require the following assumption on the VB density, which is strictly weaker than Assumption 4.

Assumption 5. *Assume that Assumption 3 applies with the function $f(\theta, \eta, t) = \mathcal{Q}(\theta|\eta)$ (no t dependence).*

Corollary 3. *Fix the quantities given in Definition 4, and let Assumptions 1 and 5 hold. Let $g(\eta) : \Omega_\eta \mapsto \mathbb{R}$ denote a continuously differentiable real-valued function of interest. Define the "influence function" $\Psi : \Omega_\theta \mapsto \mathbb{R}$:*

$$\Psi(\theta) := -\left.\frac{dg(\eta)}{d\eta^T}\right|_{\hat{\eta}} \hat{H}^{-1}\mathcal{S}(\theta|\hat{\eta})\mathcal{Q}(\theta|\hat{\eta}). \tag{26}$$

Then, if $\|\phi\|_\infty < \infty$, the map $t \mapsto g(\hat{\eta}(t\phi))$ is continuously differentiable at $t = 0$ with derivative

$$\left.\frac{dg(\hat{\eta}(t\phi))}{dt}\right|_0 = \int \Psi(\theta)\phi(\theta)\mu(d\theta). \tag{27}$$

Proof. It suffices to show that Assumption 5 implies Assumption 2 for the perturbation given in Definition 4 when $\|\phi\|_\infty < \infty$. Observe that $\log \tilde{\mathcal{P}}(\theta|t) = t\phi(\theta)$, so, for any $f(\theta, \eta, t)$ that satisfies the conditions of Assumption 3, $\phi(\theta)f(\theta, \eta, t) \leq \|\phi\|_\infty M(\theta)$. Therefore Assumption 3 is satisfied by $\phi(\theta)f(\theta, \eta, t)$ as well. It follows that Assumption 5 $\Rightarrow$ Assumption 4 $\Rightarrow$ Assumption 2. The form of the influence function is then given by gathering terms in Eq. 24. □

The influence function can be a useful summary of the effect of making generic changes to the prior density, as we will show in the experiments of Section 7. For visualization, it can be useful to reduce the dimension of the domain of the influence function, as we discuss in the following example.

Example 4. In the BNP example, we are perturbing each of the sticks, so we take $\theta \in [0, 1]^{K_{\max}-1}$. Formally, $\phi : [0, 1]^{K_{\max}-1} \mapsto \mathbb{R}$ can express different perturbations for the density of each of the $K_{\max} - 1$ sticks. However, when we describe "changing the stick breaking density," we mean changing each stick's prior density in the same way.

To represent perturbing all the sticks simultaneously, take some univariate perturbation $\phi_u : [0, 1] \mapsto \mathbb{R}$, and set $\phi(\nu_1, \ldots, \nu_{K_{\max}-1}) = \sum_{k=1}^{K_{\max}-1} \phi_u(\nu_k)$. By linearity of the derivative Corollary 3,

$$\left.\frac{dg(\hat{\eta}(t\phi))}{dt}\right|_0 = \int \Psi(\theta) \left(\sum_{k=1}^{K_{\max}-1} \phi_u(\nu_k)\right) d\nu_1 \ldots d\nu_{K_{\max}-1}.$$

By definition, $\mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\hat{\eta})}[\mathcal{S}(\theta|\hat{\eta})] = 0$, so $\int \Psi(\theta)\mu(d\theta) = 0$. By the mean-field assumption, $\Psi(\nu_1, \ldots, \nu_{K_{\max}-1}) = \prod_{k=1}^{K_{\max}-1} \Psi_k(\nu_k)$, where $\Psi_k(\nu_k)$ is derived from Eq. 26 but using $\theta = \nu_k$. Letting $\nu_0 \in [0, 1]$ denote the variable of integration and plugging in the preceding observations gives

$$\int \Psi(\theta)\phi(\theta)\mu(d\theta) = \int_0^1 \left(\sum_{k=1}^{K_{\max}-1} \Psi_k(\nu_0)\right) \phi_u(\nu_0)d\nu_0.$$

Thus we can say that the influence function for perturbing all the stick breaking densities simultaneously is given by the sum of the individual sticks' influence functions, which maps $[0, 1] \mapsto \mathbb{R}$. △

A.4 Worst-case prior perturbations and Fréchet differentiability

As we saw in Corollary 3, the derivative of perturbations given by Definition 4 takes the form of an integral of the influence function against the perturbation. It is natural to use the influence function to *explore* the space of priors, e.g., to find alternative priors with large influence but small $\|\phi\|_\infty$. Consider as an example the following corollary, which is the VB analogue of Gustafson [1996a, Result 11].

Corollary 4. *The "worst-case" derivative in $\mathcal{B}_\phi(\delta)$ is given by*

$$\sup_{\phi \in \mathcal{B}_\phi(\delta)} \left. \frac{dg(\hat{\eta}(t\phi))}{dt} \right|_0 = \delta \int |\Psi(\theta)| \, \mu(d\theta),$$

which is achieved at the perturbation $\phi^*(\theta) = \delta \, \mathrm{sign}\,(\Psi(\theta))$.

Proof. The result follows immediately from applying Hölder's inequality [Dudley, 2018, Theorem 5.1.2 and subsequent discussion] to Eq. 27. □

As discussed in Section 4 above, we also wish to show that the map $\phi \mapsto g(\hat{\eta}(\phi))$ is continuously Fréchet differentiable as a map from L_∞ to $\mathbb{R}^{D_\eta}$.

Theorem 5. *Let Assumptions 1 and 5 hold. Then the map $\phi \mapsto \hat{\eta}(\phi)$ is well-defined and continuously Fréchet differentiable in a neighborhood of 0 as a map from L_∞ to $\mathbb{R}^{D_\eta}$, with the derivative given in Corollary 3.*

(For a proof, see Appendix B Proof B.)

A.5 Other nonparametric prior perturbations

One might ask whether one could consider paths through the space of priors other than multiplicative, such as additive perturbations. In this section, we briefly consider a broader class of nonlinear perturbations investigated by Gustafson [1996a], of which additive and multiplicative perturbations are special cases, and show that, within this class, only multiplicative perturbations lead to Fréchet differentiable VB optima.

As in Appendix A.3, suppose we have an initial prior $\mathcal{P}_0$ and an alternative $\mathcal{P}_1$, and that we wish to parameterize a continuous path between them. Deviating from multiplicative perturbations, for some $p \in [1, \infty)$, let

$$\tilde{\mathcal{P}}(\theta | t_p) := \left((1 - t_p) \mathcal{P}_0(\theta)^{1/p} + t_p \frac{1}{p} \mathcal{P}_1(\theta)^{1/p} \right)^p. \tag{28}$$

As with Eq. 25, $\mathcal{P}(\theta|t_p = 0) = \mathcal{P}_0(\theta)$, $\mathcal{P}(\theta|t_p = 1) = \mathcal{P}_1(\theta)$, and $\mathcal{P}(\theta|t_p)$ moves continuously between the two in $t_p \in (0, 1)$. When $p = 1$, $\tilde{\mathcal{P}}(\theta|t_p)$ defines an "additive perturbation," and the limit as $p \to \infty$ gives the multiplicative perturbation of Eq. 25.

Gustafson [1996a, Result 2] states a result analogous to Lemma 1 for Eq. 28, where the $\|\cdot\|_\infty$ norm is replaced by

$$\phi(\theta|\mathcal{P}_1, p) := \mathcal{P}_1(\theta)^{1/p} - \mathcal{P}_0(\theta)^{1/p} \quad \text{and} \quad \|\phi\|_p := \left(\int |\phi(\theta)|^p \right)^{1/p}. \tag{29}$$

We refer the reader to Gustafson [1996a] for details. [3] For our present discussion, what matters is that the use of the perturbation in Eq. 28 strongly motivates the use of the norm $\|\phi(\theta|\mathcal{P}_1, p)\|_p$ when forming, for example, worst-case perturbations as in Corollary 4.

Though the $\|\phi(\theta|\mathcal{P}_1, p)\|_p$ norm does not appear to cause major difficulties for the full Bayesian posterior, [4] the $\|\phi(\theta|\mathcal{P}_1, p)\|_p$ norm is not compatible with KL divergence, in the sense that reverse KL divergence is *discontinuous* in this norm. Prior changes that are arbitrarily small according to $\|\phi(\theta|\mathcal{P}_1, p)\|_p$ can induce arbitrarily large changes in the reverse KL divergence, and so (in general) arbitrarily large changes in its optimum. The precise result is stated in Theorem 3 of Section 4 above; we now provide the proof.

Proof of Theorem 3. For the duration of the proof, we will use the shorthand that a density applied to a set represents the integral of the density over the set. For example, for a set S, $\mathcal{P}(S) = \int_S \mathcal{P}(\theta)\mu(d\theta)$.

The proof will be constructive, based on an alternative $\mathcal{P}_1(\theta)$ formed by driving $\mathcal{P}_0(\theta)$ to zero in a small interval. By making the interval narrow, we can make $\|\phi(\theta|\mathcal{P}_1, p)\|_p$ small, but by making the $\mathcal{P}_1(\theta)$ sufficiently close to zero, we can make the reverse KL divergence difference large irrespective of how narrow the interval is.

[3] Gustafson [1996a] in fact considers only pointwise positive perturbations $\phi(\theta|\mathcal{P}_1, p) > 0$, μ-almost everywhere. It is not hard to extend Lemma 1 [Gustafson, 1996a, Result 2] to permit negative perturbations, except for the fact that $\|\cdot\|_p$-neighborhoods of the zero function will always contain pointwise negative "priors." We allow for negative $\phi(\theta|\mathcal{P}_1, p)$ because otherwise $\|\cdot\|_p$ leads to counter intuitive notions of the "size" of prior perturbations, as we discuss in Appendix C, and because standard results in functional analysis used in the proof of Theorem 5 require open neighborhoods.

However, we must acknowledge that the main result of this section, Theorem 3 below, relies on the possibility that $\phi(\theta|\mathcal{P}_1, p)$ can be negative. In light of this, one might reasonably wonder whether we should in fact restrict to positive perturbations in an attempt to avoid the consequences of Theorem 3. In the view of the authors, restricting to pointwise positive perturbations is a somewhat artificial solution to a fundamental disconnect between the $\|\cdot\|_p$ norm and reverse KL divergence which we dicuss at the end of the present section. We believe that the disconnect is resolved more transparently and naturally through the use of the $\|\cdot\|_\infty$ norm and multiplicative perturbations which are allowed to be negative.

[4] Other than the fact that there exist pointwise negative priors induced by $\phi(\theta|\mathcal{P}_1, p)$ in every neighborhood of the zero function.

First, observe that

$$\mathrm{KL}\left(q(\theta)||\mathcal{P}_1(\theta)\right) - \mathrm{KL}\left(q(\theta)||\mathcal{P}_0(\theta)\right) = \mathop{\mathbb{E}}_{\mathcal{Q}(\theta)}\left[\log \frac{\mathcal{P}_1(\theta)}{\mathcal{P}_0(\theta)}\right].$$

For any set S with $\mathcal{P}_0(S) = \epsilon$, define

$$\mathcal{P}_1(\theta|S,\delta) := \frac{\delta^{\mathbb{I}(\theta\in S)}}{1+\epsilon(1-\delta)}\mathcal{P}_0(\theta).$$

Then $\mathcal{P}_1(\theta|S,\delta)$ is a valid density, and

$$\mathrm{KL}\left(q(\theta)||\mathcal{P}_1(\theta)\right) - \mathrm{KL}\left(q(\theta)||\mathcal{P}_0(\theta)\right) = \mathcal{Q}(S)\log\delta - \log\left(1+\epsilon(1-\delta)\right).$$

By Eq. 29,

$$\phi(\theta|\mathcal{P}_1,p) = \mathcal{P}_0(\theta)^{1/p}\left(\frac{\left(\delta^{1/p}\right)^{\mathbb{I}(\theta\in S)}}{\left(1+\epsilon(1-\delta)\right)^{1/p}} - 1\right) \quad \text{and}$$

$$\|\phi(\theta|\mathcal{P}_1,p)\|_p^p = \epsilon\left(\frac{\left(\delta^{1/p}\right)}{\left(1+\epsilon(1-\delta)\right)^{1/p}} - 1\right) + (1-\epsilon)\left(\frac{1}{\left(1+\epsilon(1-\delta)\right)^{1/p}} - 1\right).$$

Since μ is absolutely continuous with respect to the Lebesgue measure, there exists a sequence $\epsilon_n \to 0$ with $\epsilon_n > 0$ and a sequence of corresponding sets S_n such that $\mathcal{P}_0(S_n) = \epsilon_n$. (See Lemma 6 for a proof of this fact, which is a straightforward consequence of Nielsen [1997, Proposition 15.5] and the continuity of the Lebesgue measure.) Since $\mathcal{Q}(\theta) > 0$ on Ω_θ, $\mathcal{Q}(S_n) > 0$ for all n. Since $\mathrm{KL}\left(\mathcal{Q}(\theta)||\mathcal{P}_0(\theta)\right)$ is finite, we must have $\lim_{n\to}\mathcal{Q}(S_n) = 0$.

Take $\delta_n = \exp(-1/(\mathcal{Q}(S_n)^2))$, and take $\mathcal{P}_1(\theta) = \mathcal{P}_1(\theta|S_n,\delta_n)$. Then $\epsilon_n(1-\delta_n) \to 0$, and $\mathcal{Q}(S_n)\log\delta_n = -1/\mathcal{Q}(S_n)$, so

$$\begin{aligned}\left|\mathrm{KL}\left(q(\theta)||\mathcal{P}_1(\theta|S_n,\delta_n)\right) - \mathrm{KL}\left(q(\theta)||\mathcal{P}_0(\theta)\right)\right| \to \infty, \quad &\text{but}\\ \|\phi(\theta|\mathcal{P}_1(\cdot|S_n,\delta_n),p)\|_p^p \to 0.&\end{aligned}$$

Thus, for sufficiently large n, the conclusion follows. □

B Detailed Proofs

In this section, we provide detailed proofs for results stated above.

A standard consequence of the dominated convergence theorem is the ability to exchange integration and differentiation. Since we will use this result frequently, we state it here in our own notation as Theorem 6.

Theorem 6. *[Billingsley, 1986, Theorem 16.8] Let μ be sigma-finite measure on Ω_θ, and let $S_t \subseteq \mathbb{R}$. Let $f : \Omega_\theta \times S_t \mapsto \mathbb{R}$.*

If there exists a function $M(\theta)$ with $\int M(\theta)\mu(d\theta) < \infty$ such that $|f(\theta,t)| \leq M(\theta)$, μ-almost surely, for all $t \in S_t$, then the map $t \mapsto \int f(\theta,t)\mu(d\theta)$ is continuous.

Further, suppose that the derivative $\frac{\partial f(\theta,t)}{\partial t}\Big|_t$ exist μ-almost surely for $t \in S_t$. If there exists an $M'(\theta)$ such that $\int M'(\theta)\mu(d\theta) < \infty$ and $\left|\frac{\partial f(\theta,t)}{\partial t}\Big|_t\right| \leq M'(\theta)$, μ-almost surely and for all $t \in S_t$, then

$$\frac{\partial \int f(\theta,t)\mu(d\theta)}{\partial t}\bigg|_t = \int \frac{\partial f(\theta,t)}{\partial t}\bigg|_t \mu(d\theta).$$

Lemma 2. *Under Assumption 3, at any $\eta, t \in \mathcal{B}_\eta \times \mathcal{B}_t$, we can exchange the order of integration and differentiation in $\int f(\theta,\eta,t)\mu(d\theta)$ for the derivatives $\partial/\partial\eta$, $\partial^2/\partial\eta^2$, and $\partial^2/\partial\eta\partial t$.*

Proof. Let η_d denote the $d-$th entry of the vector η. Then

$$\begin{aligned}
|\partial f(\theta,\eta,t)/\partial\eta_d| &\leq \|\partial f(\theta,\eta,t)/\partial\eta\|_2, \\
\left|\partial^2 f(\theta,\eta,t)/\partial\eta_d\partial t\right| &\leq \|\partial f(\theta,\eta,t)/\partial\eta\partial t\|_2, \text{ and} \\
\left|\partial^2 f(\theta,\eta,t)/\partial\eta_{d_1}\partial\eta_{d_2}\right| &\leq \left\|\partial f(\theta,\eta,t)/\partial\eta\partial\eta^T\right\|_2.
\end{aligned}$$

The conclusion follows by repeatedly applying Theorem 6 to the components of the derivatives.

□

Lemma 3. *Under Assumption 2, the map $\eta, t \mapsto \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\log\tilde{\mathcal{P}}(\theta|t)\right]$ has continuous partial derivatives $\partial/\partial\eta$, $\partial^2/\partial\eta^2$, and $\partial^2/\partial\eta\partial t$ at all $\eta, t \in \mathcal{B}_\eta \times \mathcal{B}_t$. Furthermore,*

$$\frac{\partial \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\log\tilde{\mathcal{P}}(\theta|t)\right]}{\partial\eta}\bigg|_\eta = \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\mathcal{S}(\theta|\eta)\log\tilde{\mathcal{P}}(\theta|t)\right] \tag{30}$$

$$\frac{\partial^2 \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\log\tilde{\mathcal{P}}(\theta|t)\right]}{\partial\eta\partial t}\bigg|_{\eta,t} = \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\mathcal{S}(\theta|\eta)\,\frac{\partial\log\tilde{\mathcal{P}}(\theta|t)}{\partial t}\bigg|_t\right]. \tag{31}$$

Proof. We can write

$$R(a,b) := \frac{a}{b} \quad \Rightarrow \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}[\psi(\theta,t)] = R\left(\int\tilde{\mathcal{Q}}(\theta|\eta)\psi(\theta,t)\mu(d\theta), \int\tilde{\mathcal{Q}}(\theta|\eta)\mu(d\theta)\right).$$

If necessary, we can shrink $\mathcal{B}_\eta$ so that the denominator $\int\tilde{\mathcal{Q}}(\theta|\eta)\mu(d\theta)$ is bounded below by a positive constant for all $\eta \in \mathcal{B}_\eta$. With the denominator strongly positive, $R(a,b)$ is is a continuously differentiable function to all orders for all $t, \eta \in \mathcal{B}_t \times \mathcal{B}_\eta$. The desired results follow from Assumption 2 by the chain rule.

□

Proof of Theorem 4. By Assumption 1 (Item (1)) and Lemma 3, $\eta \mapsto \mathrm{KL}(\eta, t)$ is continuously differentiable for all $\eta, t \in \mathcal{B}_\eta \times \mathcal{B}_t$. So, for all $t \in \mathcal{B}_t$, the optimal $\hat{\eta}(t)$ satisfies the first order condition:

$$\left.\frac{\partial \mathrm{KL}(\eta, t)}{\partial \eta}\right|_{\hat{\eta}(t),t} = \left.\frac{\partial \mathrm{KL}(\eta)}{\partial \eta}\right|_{\hat{\eta}(t)} + \left.\frac{\partial \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\log \tilde{\mathcal{P}}(\theta|t)\right]}{\partial \eta}\right|_{\hat{\eta}(t)} = 0 \tag{32}$$

We wish to apply the implicit function theorem [Krantz and Parks, 2012, Theorem 3.3.1] to the estimating equation defined by Eq. 32. Again, by Assumption 1 (Item (1)) and Lemma 3, the estimating equation given is continuously differentiable in both η and t. The Jacobian of the estimating equation is nonsingular by Assumption 1 (Item (2)), and valid in an open ball by Assumption 1 (Item (3)). Finally, the form of the derivative is given by Krantz and Parks [2012, Theorem 3.3.1], together with Eq. 31 of Lemma 3.

For convenience, Table B shows the correspondence between our notation and that of Krantz and Parks [2012, Theorem 3.3.1].

Krantz & Parks notation	Our notation
$\Phi(x)$	$\mathrm{KL}(\eta, t)$
Q	1
M	D_η
U	$\mathcal{B}_\eta \times \mathcal{B}_t$
W	$\mathcal{B}_t$
$x_1, \ldots, x_Q$	t
$x_{Q+1}, \ldots, x_N$	η
$f_1(x_a), \ldots, f_M(x_a)$	$\hat{\eta}(t)$

□

Proof of Lemma 1. Let μ and μ' denote two mutually absolutely continuous candidate dominating measures for $\mathcal{P}_0$, with respective densities (Radon-Nikodym derivatives) $\mathcal{P}_0(\theta)$ and $\mathcal{P}_0'(\theta)$. Let the respective densities of the measure $\mathcal{P}$ be denoted $\mathcal{P}_1(\theta)$ and $\mathcal{P}_1'(\theta)$ as well. Let $R(\theta) = \left.\frac{d\mu}{d\mu'}\right|_\theta$ denote the Radon-Nikodym derivative of μ with respect to μ', and note that $\mathcal{P}_0'(\theta) = R(\theta)\mathcal{P}_0(\theta)$ and $\mathcal{P}_1'(\theta) = R(\theta)\mathcal{P}_1(\theta)$.

We have that the perturbations for μ and μ' are given respectively by

$$\begin{aligned}
\phi(\theta|\beta, \mathcal{P}_1) &= \log \mathcal{P}_1(\theta) - \log \mathcal{P}_0(\theta) + \log \beta \\
\phi'(\theta|\beta, \mathcal{P}_1') &= \log \mathcal{P}_1'(\theta) - \log \mathcal{P}_0'(\theta) + \log \beta \\
&= \log \mathcal{P}_1(\theta) - \log R(\theta) - \log \mathcal{P}_0(\theta) + \log R(\theta) + \log \beta \\
&= \phi(\theta|\beta, \mathcal{P}_1).
\end{aligned}$$

It follows that $\|\phi(\cdot|\beta, \mathcal{P}_1)\|_\infty = \|\phi'(\cdot|\beta, \mathcal{P}_1')\|_\infty$.

Next, let $\tau := \tau(\theta)$ be an invertible transformation with Jacobian $J(\theta) := \det\left(\frac{d\tau}{d\theta^T}\big|_\theta\right)$. For the dominating measure μ, let $\mathcal{P}_0(\theta)$ and $\mathcal{P}_1(\theta)$ denote the densities of θ and $\mathcal{P}_0'(\tau)$ and $\mathcal{P}_1'(\tau)$ denote the densities of τ. The desired result follows by the exact same formal argument as for the change of measure, except with $J(\theta)\mu(d\theta)$ and $\mu(d\tau)$ taking the place of $R(\theta)\mu(d\theta)$ and $\mu'(d\theta)$, respectively.

We now prove that ϕ gives rise to valid priors when $\|\phi\|_\infty < \infty$. Since the exponential function is positive, for any $\phi(\theta)$,

$$\tilde{\mathcal{P}}(\theta|\phi) = \mathcal{P}_0(\theta)\exp(\phi(\theta)) > 0,$$

μ-almost everywhere. Furthermore, since $\int \mathcal{P}_0(\theta)\lambda(d\theta) = 1$,

$$\exp(-\|\phi\|_\infty) \leq \int \mathcal{P}_0(\theta)\exp\left(\phi(\theta)\right)\mu(d\theta) \leq \exp(\|\phi\|_\infty).$$

so that $0 < \int \tilde{\mathcal{P}}(\theta|\phi)\mu(d\theta) < \infty$ whenever $\|\phi\|_\infty < \infty$. □

Lemma 4. *Under Definition 4, Assumption 5 implies Assumption 2 when* $\|\phi\|_\infty < \infty$.

Proof. Since $\tilde{\mathcal{Q}}(\theta|\eta)\phi(\theta) \leq \tilde{\mathcal{Q}}(\theta|\eta)\delta$, and $\tilde{\mathcal{Q}}(\theta|\eta)$ satisfies Assumption 3 with some $M(\theta)$ by Assumption 5, we can satisfy Assumption 3 for $\tilde{\mathcal{Q}}(\theta|\eta)\phi(\theta)$ with $\max\{1,\delta\}M(\theta)$. Finally, Lemma 2 implies that Assumption 2 is satisfied. □

Lemma 5. *Under Assumption 5, the map* $\eta, \phi \mapsto \partial \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}[\phi(\theta)]/\partial\eta$ *is continuously Fréchet differentiable as a map from* $\mathbb{R}^{D_\eta} \times L_\infty \mapsto \mathbb{R}^{D_\eta}$.

Proof. The map $\eta, \phi \mapsto \partial \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}[\phi(\theta)]/\partial\eta$ is a map from the Banach space $\mathbb{R}^{D_\eta} \times L_\infty$ into the Banach space $\mathbb{R}$. Let us take the L2 norm $\|\cdot\|_2$ on $\mathbb{R}^{D_\eta}$ and $\mathbb{R}$. Let $\mathcal{B}$ denote the ball $\mathcal{B}_\eta \times \{\phi : \|\phi\|_\infty < \delta\}$ for some $\delta > 0$. Let $\mathcal{L}$ denote a linear operator from $\mathcal{B}$ to $\mathbb{R}^{D_\eta}$, and define the dual norm

$$\|\mathcal{L}\|^* := \sup_{\Delta\eta:\|\eta\|_2\leq 1} \sup_{\Delta\phi:\|\phi\|_\infty\leq 1} \|\mathcal{L}(\Delta\eta, \Delta\phi)\|_2 .$$

Formally, $\Delta\eta$ and $\Delta\phi$ are members of $\mathbb{R}^{D_\eta}$ and L_∞ respectively, but in the preceding display they can be thought of as directions on which the linear operator $\mathcal{L}$ operates.

Observe that the directional derivatives are linear operators, and so $\|\cdot\|^*$ defines a norm on the space of linear operators. We will prove Fréchet differentiability using the fact that a functional is Fréchet differentiable if its directional derivatives are continuous in $\|\cdot\|^*$ as a function of the location at which they are evaluated (Zeidler [1986, Proposition 4.8(c)], Averbukh and Smolyanov [1967, Corollary 1.4] and [Reeds, 1976, Appendix A]). Further, it suffices by Zeidler [1986, Proposition 4.14(c)] to show that the partial derivatives with respect to η and ϕ are continuously Fréchet to show that the joint map is continuously Fréchet differentiable.

First, consider the partial derivative with respect to η. Observe that, by Lemma 4, Lemma 3 applies with $\log \tilde{\mathcal{P}}(\theta|t) = \phi(\theta)$ (no t dependence). Consequently the map $\eta \mapsto \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}[\phi(\theta)]$ is twice continuously differentiable. The linear operator corresponding to the directional derivative in the $\Delta\eta$ direction is given by

$$\mathcal{L}_\eta(\Delta\eta, \Delta\phi) = \left.\frac{\partial^2 \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}[\phi(\theta)]}{\partial\eta\partial\eta^T}\right|_\eta \Delta\eta,$$

with no dependence on $\Delta\phi$. Define for the moment the the $D_\eta \times D_\eta$ matrix $\mathscr{H}(\eta, \phi) := \partial^2 \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}[\phi(\theta)] / \partial\eta\partial\eta^T$. Then the dual norm of the derivative is simply the operator norm of $\mathscr{H}$, i.e., $\|\mathcal{L}_\eta\|^* = \|\mathscr{H}(\eta, \phi)\|_{op}$. Thus we must show that $\|\mathscr{H}(\eta, \phi)\|_{op}$ is continuous in η, ϕ. For any η', ϕ' and η'', ϕ'' in $\mathcal{B}_\eta \times \mathcal{B}_\phi(\delta)$,

$$\begin{aligned}&\|\mathscr{H}(\eta', \phi') - \mathscr{H}(\eta'', \phi'')\|_{op} \\ &\quad\leq \|\mathscr{H}(\eta', \phi') - \mathscr{H}(\eta', \phi'')\|_{op} + \|\mathscr{H}(\eta', \phi'') - \mathscr{H}(\eta'', \phi'')\|_{op}.\end{aligned}$$

For the first term in the preceding display, for all η',

$$\|\mathscr{H}(\eta', \phi') - \mathscr{H}(\eta', \phi'')\|_{op} \leq \left.\frac{\partial^2 \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}[1]}{\partial\eta\partial\eta^T}\right|_{\eta'} \|\phi' - \phi''\|_\infty \Rightarrow$$

$$\lim_{\phi' \to \phi''} \|\mathscr{H}(\eta', \phi') - \mathscr{H}(\eta', \phi'')\|_{op} = 0.$$

For the second term, by Lemma 3, for all ϕ'',

$$\lim_{\eta' \to \eta''} \|\mathscr{H}(\eta', \phi'') - \mathscr{H}(\eta'', \phi'')\|_{op} = 0.$$

It follows that $\|\mathscr{H}(\eta, \phi)\|_{op}$ is continuous in η, ϕ, and so the partial derivative with respect to η is a continuous Fréchet derivative.

Next, we consider the partial derivative with respect to ϕ. By Eq. 30, we can write

$$\left.\frac{\partial \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}[\phi(\theta)]}{\partial\eta}\right|_\eta = \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}[\mathcal{S}(\theta|\eta)\phi(\theta)].$$

Since this expression is linear in ϕ, the linear operator for the partial derivative with respect to ϕ is given by

$$\mathcal{L}_\phi(\Delta\eta, \Delta\phi) = \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}[\mathcal{S}(\theta|\eta)\Delta\phi(\theta)],$$

with no dependence on $\Delta\eta$.

In order to be a valid partial derivaive, we must verify that $\mathcal{L}_\phi$ is a bounded linear operator. Boundedness follows from Hölder's inequality and Assumption 5 since

$$\begin{aligned}\sup_{\Delta\phi:\|\Delta\phi\|_\infty\leq 1}\|\mathcal{L}_\phi(\Delta\eta,\Delta\phi)\|_2 &\leq \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\|\mathcal{S}(\theta|\eta)\|_1\right]\|\Delta\phi\|_\infty \\ &\leq \sqrt{D_\eta}\mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\|\mathcal{S}(\theta|\eta)\|_2\right] \\ &\leq \sqrt{D_\eta}\int M(\theta)\mu(d\theta) < \infty.\end{aligned}$$

Similarly, the dual norm of the ϕ partial derivative is given by

$$\|\mathcal{L}_\phi\|^* = \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\|\mathcal{S}(\theta|\eta)\|_1\right].$$

We thus need to show that $\eta \mapsto \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\|\mathcal{S}(\theta|\eta)\|_1\right]$ is a continuous function of η (there is no ϕ dependence). To show this, observe that

$$\mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\|\mathcal{S}(\theta|\eta)\|_1\right] = \frac{\int \tilde{\mathcal{Q}}(\theta|\eta)\,\|\mathcal{S}(\theta|\eta)\|_1\,\mu(d\theta)}{\int \tilde{\mathcal{Q}}(\theta|\eta)\mu(d\theta)}. \tag{33}$$

By Assumption 5, we have that there exists a finitely integrable envelope function $M(\theta)$ such that, for all $\eta \in \mathcal{B}_\eta$,

$$\begin{aligned}\tilde{\mathcal{Q}}(\theta|\eta) &\leq M(\theta) \quad \text{and} \\ \tilde{\mathcal{Q}}(\theta|\eta)\,\|\mathcal{S}(\theta|\eta)\|_1 &\leq \sqrt{D_\eta}\tilde{\mathcal{Q}}(\theta|\eta)\,\|\mathcal{S}(\theta|\eta)\|_2 \leq M(\theta).\end{aligned}$$

Therefore, by the dominated convergence theorem, we can exchange limits and integrals in the numerator and denominator of Eq. 33. It follows that, for any η' and η'' in $\mathcal{B}_\eta$,

$$\begin{aligned}\lim_{\eta'\to\eta''}\left|\int \tilde{\mathcal{Q}}(\theta|\eta')\mu(d\theta) - \int \tilde{\mathcal{Q}}(\theta|\eta'')\mu(d\theta)\right| &\leq \lim_{\eta'\to\eta''}\int\left|\tilde{\mathcal{Q}}(\theta|\eta') - \tilde{\mathcal{Q}}(\theta|\eta'')\right|\mu(d\theta) \\ &= \int \lim_{\eta'\to\eta''}\left|\tilde{\mathcal{Q}}(\theta|\eta') - \tilde{\mathcal{Q}}(\theta|\eta'')\right| = 0.\end{aligned}$$

Thus the numerator of Eq. 33 is continuous in η. The denominator of Eq. 33 is also continuous by an analogous argument. Since the denominator of Eq. 33 is bounded away from zero, $\mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\|\mathcal{S}(\theta|\eta)\|_1\right]$ is a continuous composition of continuous functions, and itself continuous. It follows that the ϕ partial derivative is continuously Fréchet differentiable.

Since its partial derivatives are continuous, it follows by Zeidler [1986, Proposition 4.14(c)] that the joint map $\eta, \phi \mapsto \partial \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\phi(\theta)\right]/\partial\eta$ is continously Fréchet differentiable.

□

Proof of Theorem 5. Recall that, by Lemma 4, Lemma 3 applies with $\log \tilde{\mathcal{P}}(\theta|t) = \phi(\theta)$ (no t dependence). Therefore, as in the proof of Theorem 4, for any $\phi \in \mathcal{B}_\phi(\delta)$, $\hat{\eta}(\phi)$ satisfies the first-order condition

$$\left.\frac{\partial \mathrm{KL}\left(\eta\right)}{\partial \eta}\right|_{\hat{\eta}(\phi)} + \left.\frac{\partial \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)}\left[\phi(\theta)\right]}{\partial \eta}\right|_{\hat{\eta}(\phi)} = 0. \tag{34}$$

As in the proof of Theorem 4, we wish to employ an implicit function theorem, but this time for general Banach spaces. We will us Zeidler [1986, Theorem 4.B].

First, Zeidler [1986, Chapter 4 Condition 21b] holds since $\hat{H}$ is invertible by Assumption 1 (Item (2)). So we satisfy conditions (i), (ii), and (iii) of Zeidler [1986, Theorem 4.B(c)], giving that the function $\hat{\eta}(\phi)$ exists.

Moreover, by Assumption 1 (Item (1)) and Lemma 5, the estimating equation Eq. 34 is continuously Freéchet differentiable (C^1 in the notation of Zeidler) in a neighborhood of $\hat{\eta}, 0$. It follows from Zeidler [1986, Theorem 4.B(d)], $\hat{\eta}(\phi)$ is also continuously Fréchet differentiable. □

Lemma 6. *If μ is absolutely continuous with respect to the Lebesgue measure, and $\mathcal{P}_0$ is a probability measure with a density relative to μ, then there exists a sequence $\epsilon_n \to 0$ with $\epsilon_n > 0$ and a sequence of corresponding sets S_n such that $\mathcal{P}_0(S_n) = \epsilon_n$.*

Proof. Let $\epsilon'_n = n^{-1}$. Since $\mathcal{P}_0 \ll \mu \ll \lambda$ (where λ is the Lebesgue measure), by applying Nielsen [1997, Proposition 15.5], for each n there exists a δ'_n such that, for any measureable set A with $\mu(A) < \delta'_n$, $\mathcal{P}_0(A) < \epsilon'_n$. Again applying Nielsen [1997, Proposition 15.5], there similarly exists a δ_n such that for any measureable set A with $\lambda(A) < \delta_n$, $\mu(A) < \delta'_n \Rightarrow \mathcal{P}_0(A) < \epsilon'_n$.

For each n, partition Ω_θ into a countable number of sets A_m such that $\sum_m \lambda(A_m) = 1$ and $\lambda(A_m) < \delta_n$. (This is possible by dividing Ω_θ into sufficiently small rectangles, for example.) Then $\mathcal{P}_0(A_m) < \epsilon'_n$ for all m. Since $\mathcal{P}_0$ is a probability measure, $\sum_m \mathcal{P}_0(A_m) = 1$, so there must exist at least $1/\epsilon'_n$ indices m' such that $\mathcal{P}_0(A_{m'}) > 0$. Take any such m' and let $\epsilon_n = \mathcal{P}_0(A_{m'})$ and $S_n = A_{m'}$. □

Lemma 7. *Let μ denote the Lebesgue measure on $\mathbb{R}$. Let σ denote the standard deviation of a normal distribution, let η denote a continuously differentiable function of the normal distribution's natural parameters, let $\mathcal{N}(\theta|\eta)$ denote a normal density with respect to μ, and let $\mathcal{B}_\eta$ denote an open ball in $\mathbb{R}^2$ such that $0 < \sigma < \infty$ for all $\eta \in \mathcal{B}_\eta$. Let $\mathcal{B}_t$ denote an open ball in $\mathbb{R}$, and let $\psi(\theta, t)$ be a function such that $\theta \mapsto \psi(\theta, t)$ is μ-measurable for all $t \in \mathcal{B}_t$.*

If there exists a constant $C > 0$ such that

$$\sup_{t \in \mathcal{B}_t} |\psi(\theta, t)| \leq C \exp(|\theta|) \quad \textit{and} \quad \sup_{t \in \mathcal{B}_t} \left|\frac{\partial \psi(\theta, t)}{\partial t}\right| \leq C \exp(|\theta|),$$

then one can exchange the order of expectation and differentation in the expression $\mathop{\mathbb{E}}_{\mathcal{N}(\theta|\eta)}[\psi(\theta,t)]$ *for the derivatives* $\partial/\partial\eta$, $\partial^2/\partial\eta\partial\eta$, *and* $\partial^2/\partial\eta\partial t$, *eavluated at any* $t \in \mathcal{B}_t$ *and* $\eta \in \mathcal{B}_\eta$.

Proof. For the moment, let η denote the exponential family natural parameters of the normal distribution. By properties of the exponential family,

$$\left.\frac{\partial \log \tilde{\mathcal{Q}}(\theta|\eta)}{\partial \eta}\right|_\eta = (\theta, \theta^2)^T \quad \text{and} \quad \left.\frac{\partial^2 \log \tilde{\mathcal{Q}}(\theta|\eta)}{\partial \eta \partial \eta^T}\right|_\eta = 0_{2\times 2} \Rightarrow$$
$$\left\| \left.\frac{\partial \log \tilde{\mathcal{Q}}(\theta|\eta)}{\partial \eta}\right|_\eta \right\|_2^2 = \theta^2 + \theta^4 \quad \text{and} \quad \left\| \left.\frac{\partial^2 \log \tilde{\mathcal{Q}}(\theta|\eta)}{\partial \eta \partial \eta^T}\right|_\eta \right\|_2 = 0.$$

Let $\overline{\mathcal{B}}_\eta$ denote the closure of $\mathcal{B}_\eta$, and let

$$\eta^* := \mathop{\text{argmax}}_{\eta \in \overline{\mathcal{B}}_\eta} \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta)} [\exp(|\theta|)] .$$

By standard properties of the normal and the boundedness of $\sigma(\eta)$, the right hand side of the preceding display is finite. Then

$$\begin{aligned}\int \mathcal{Q}(\theta|\eta)\psi(\theta,t)\mu(d\theta) &\leq \left(\sup_\theta \sup_{t\in\mathcal{B}_t} |\psi(\theta,t)| \exp(-|\theta|)\right) \int \mathcal{Q}(\theta|\eta)\exp(\theta)\mu(d\theta) \\ &\leq C \mathop{\mathbb{E}}_{\mathcal{Q}(\theta|\eta^*)} [\exp(|\theta|)] . \quad (C \text{ does not depend on } \eta, t)\end{aligned}$$

Therefore, for Assumption 3, we can take $M(\theta) \propto \mathcal{Q}(\theta|\eta^*)\exp(|\theta|)$. The other terms follow similarly, since each multiplier of $\mathcal{Q}(\theta|\eta)$ is dominated by $\exp(-|\theta|)$. The final $M(\theta)$ simply takes the largest of the five constants.

Finally, if $\tilde{\eta}$ is a twice-continuously differentiable function of the natural parameters η (e.g the mean and variance), then the derivatives with respect to $\tilde{\eta}$ are equal to the derivatives with respect to η times bounded (on $\mathcal{B}_\eta$) functions of η that do not depend on θ. Thus a constant multiple of $M(\theta)$ will bound the new derivatives. □

C Positive Perturbations Are Counterintuitive

The following example illustrates how, by requiring pertubations to be positive, one can induce counterintuitive notions of the "size" of a perturbation that ablates prior mass.

Example 5. Take μ to be the Lebesgue measure on $[0,1]$. Let $\mathcal{P}_0(\theta) = \mathbb{I}(0 \leq \theta \leq 1)$. For some $\delta > 0$ and $0 < \epsilon \ll 1$, let

$$\mathcal{P}_1(\theta) := \left(\frac{1-\delta\epsilon}{1-\epsilon}\right) \mathbb{I}(\epsilon \leq \theta \leq 1) + \delta \mathbb{I}(0 \leq \theta \leq \epsilon) .$$

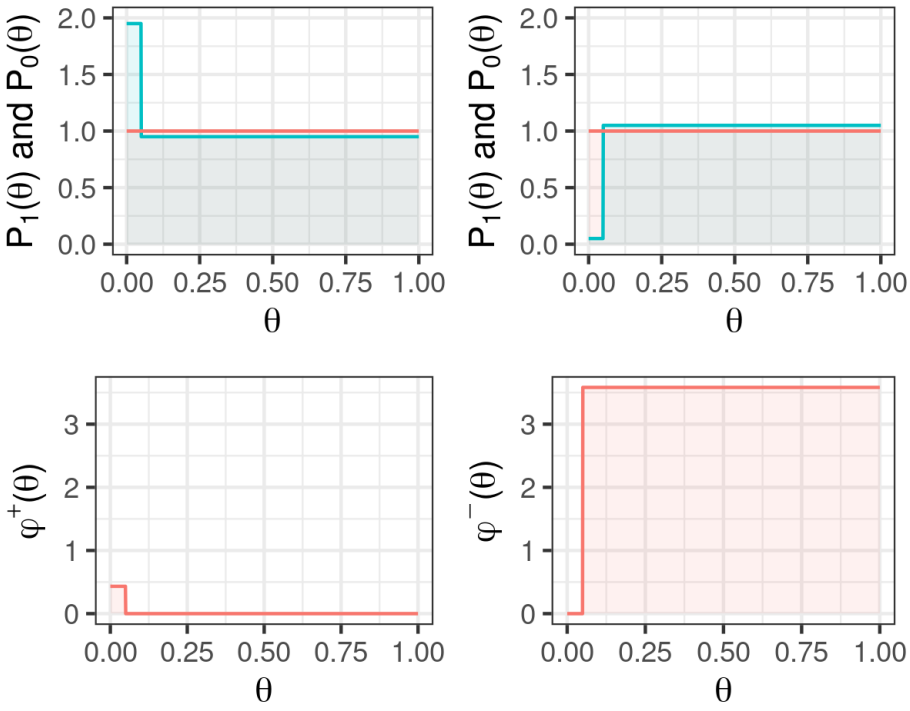


Figure 11: A plot of the perturbations from Example 5 with $p = 2$ and $\epsilon = 0.05$. Positive ϕ can only add mass, so to remove a small amount of mass requires adding mass everywhere else and re-normalizing, resulting in a large perturbation according to $\|\cdot\|_p$.

We can use Eq. 28 to give $\tilde{\mathcal{P}}(\theta|t_p = 1) = \mathcal{P}_1(\theta)$ by using, for any $\alpha > 0$,

$$\phi(\theta) = \left(\alpha\left(\frac{1-\delta\epsilon}{1-\epsilon}\right)^{1/p} - 1\right)\mathbb{I}\left(\epsilon \leq \theta \leq 1\right) + \left(\alpha\delta^{1/p} - 1\right)\mathbb{I}\left(0 \leq \theta \leq \epsilon\right).$$

It follows that

$$\|\phi\|_p = \left(\alpha\left(\frac{1-\delta\epsilon}{1-\epsilon}\right)^{1/p} - 1\right)(1-\epsilon) + \left(\alpha\delta^{1/p} - 1\right)\epsilon.$$

For ϕ to be positive, we require

$$\alpha^p \geq \frac{1-\epsilon}{1-\delta\epsilon} \quad \text{and} \quad \alpha^p \geq \frac{1}{\delta}.$$

First, let us consider adding a small amount of prior mass, taking $\delta = 2 - \epsilon$; let the corresponding perturbation be ϕ^+. For $\delta > 1$, then we achieve $\phi \geq 0$ by taking

$\alpha^p = \frac{1-\epsilon}{1-\delta\epsilon}$. Using the fact that $\epsilon \ll 1$ and keeping only leading-order terms,

$$\begin{aligned}\frac{1-\epsilon}{1-\delta\epsilon} &\approx (1-\epsilon)(1+\delta\epsilon)\\ &\approx 1+(\delta-1)\epsilon\\ &\approx 1+\epsilon,\end{aligned}$$

so

$$\begin{aligned}\left\|\phi^+\right\|_p &= \left(\alpha\delta^{1/p}-1\right)\epsilon\\ &\approx \left(\left((1+\epsilon)\,(2-\epsilon)\right)^{1/p}-1\right)\epsilon\\ &\approx \left(2^{1/p}-1\right)\epsilon.\end{aligned}$$

Next, consider removing the same amount of mass with the symmetric change $\delta = \epsilon$, letting ϕ^- be the corresponding perturbation. Then we can ensure that $\phi(\theta) \geq 0$ with $\alpha^p \geq \epsilon^{-1}$, and $\epsilon \ll 1$ gives

$$\frac{1-\delta\epsilon}{1-\epsilon} \approx 1-\epsilon,$$

and

$$\begin{aligned}\left\|\phi^-\right\|_p &= \left(\alpha\left(\frac{1-\delta\epsilon}{1-\epsilon}\right)^{1/p}-1\right)(1-\epsilon)\\ &\approx \left(\left(\frac{1-\epsilon}{\epsilon}\right)^{1/p}-1\right)(1-\epsilon)\\ &\approx \left(\frac{1}{\epsilon}\right)^{1/p}.\end{aligned}$$

Since ϵ is small, $\left\|\phi^-\right\|_p \approx \left(\frac{1}{\epsilon}\right)^{1/p} \gg \left\|\phi^+\right\|_p \approx \left(2^{1/p}-1\right)\epsilon$, despite the two perturbations respectively removing and adding the same amount of arbitrarily small probability mass.

△

D Computational details

D.1 The optimal local parameters

In all models we consider, the optimal local variational parameters $\hat{\eta}_\ell$ can be written as a closed-form function of the global variational parameters η_γ. Let $\hat{\eta}_\ell(\eta_\gamma; t)$ denote this mapping; that is,

$$\hat{\eta}_\ell(\eta_\gamma; t) := \operatorname*{argmin}_{\eta_\ell} \mathrm{KL}\left((\eta_\gamma, \eta_\ell), t\right).$$

The next example details this mapping for the Gaussian mixture model.

Example 6 (Optimalility of η_ℓ in a GMM)**.** Recall that under our truncated variational approximation, the cluster assignment z_n is a discrete random variable over $K_{\max}$ categories.

Let η_{z_n} be the categorical parameters in its exponential family natural parameterization. That is, we let $\eta_{z_n} = (\rho_{n1}, \rho_{n2}, ..., \rho_{n(K_{\max}-1)})$ be an unconstrained vector in $\mathbb{R}^{K_{\max}-1}$; in this parameterization, the assignment probabilities are

$$p_{nk} := \mathop{\mathbb{E}}_{\mathcal{Q}(z_n|\eta_z)}[z_{nk}] = \frac{\exp(\rho_{nk})}{1 + \sum_{k'=1}^{K_{\max}-1} \exp(\rho_{nk})}$$

We use the exponential family parameterization because we require the optimal variational parameters $\hat{\eta}$ to be interior to Ω_η in Theorem 4. In the mean parameterization, $\sum_{k=1}^{K_{\max}} p_{nk} = 1$, so the optimal mean parameters $\hat{p}_n$ cannot be interior to $\Delta^{K_{\max}-1}$. On the other hand, η_{z_n} as defined is unconstrained in $\mathbb{R}^{K_{\max}-1}$.

Fixing $\mathcal{Q}(\beta|\eta_\beta)$ and $\mathcal{Q}(\nu|\eta_\nu)$, the optimal $\hat{\eta}_{z_n}$ must satisfy

$$\begin{aligned} &\mathcal{Q}(z_n|\hat{\eta}_{z_n}) \propto \exp\left(\tilde{\rho}_{nk}\right) \\ &\quad \text{where} \quad \tilde{\rho}_{nk} := \mathop{\mathbb{E}}_{\mathcal{Q}(\beta,\nu|\eta)}\left[\log \mathcal{P}(x_n|\beta_k) + \log \pi_k\right]. \end{aligned}$$

See Bishop [2006] and Blei et al. [2017] for details. To satisfy this optimality condition, we set the optimal $\hat{\eta}_{z_n}$ to be

$$\hat{\eta}_{z_n} = \left(\log \frac{\tilde{\rho}_{n1}}{\tilde{\rho}_{nK_{\max}}}, \log \frac{\tilde{\rho}_{n2}}{\tilde{\rho}_{nK_{\max}}}, \ldots, \log \frac{\tilde{\rho}_{n(K_{\max}-1)}}{\tilde{\rho}_{nK_{\max}}}\right).$$

Thus, as long as the expectations $\tilde{\rho}_{nk}$ can be provided as a closed-form function of (η_β, η_ν), the optimal $\hat{\eta}_{z_n}$ can be also be set in closed-form as a function of (η_β, η_ν). △

D.2 More details on computing and inverting the Hessian

We fill in more details for the efficient computation of the Hessian outlined in Section 6.

We start from our formula in Eq. 18,

$$\left.\frac{d\hat{\eta}(t)}{dt}\right|_{t=0} = -\begin{pmatrix} H_{\gamma\gamma} & H_{\gamma\ell} \\ H_{\ell\gamma} & H_{\ell\ell} \end{pmatrix}^{-1} \begin{pmatrix} \hat{J}_\gamma \\ 0 \end{pmatrix},$$

and an application of the Schur complement gives

$$\left.\frac{d\hat{\eta}(t)}{dt}\right|_{t=0} = -\begin{pmatrix} I_{\gamma\gamma} \\ H_{\ell\ell}^{-1} H_{\ell\gamma} \end{pmatrix} \left(H_{\gamma\gamma} - H_{\gamma\ell} H_{\ell\ell}^{-1} H_{\ell\gamma}\right)^{-1} \hat{J}_\gamma,$$

where $I_{\gamma\gamma}$ is the identity matrix with the same dimension as η_γ. Specifically, observe that the sensitivity of the global parameters is given by

$$\left.\frac{d\hat{\eta}_\gamma(t)}{dt}\right|_{t=0} = -\hat{H}_\gamma^{-1} \hat{J}_\gamma \quad \text{where} \quad \hat{H}_\gamma := \left(H_{\gamma\gamma} - H_{\gamma\ell} H_{\ell\ell}^{-1} H_{\ell\gamma}\right),$$

In our model, $H_{\ell\ell}$ is sparse, and the size of $H_{\gamma\gamma}$ does not grow with N. Thus, each term of $\hat{H}_\gamma$ can be tractably computed, stored in memory, and inverted, even on very large datasets.

One can derive the exact same identity using the optimality of $\hat{\eta}_\ell(\eta_\gamma)$. By applying the chain rule, one can verify that

$$\hat{H}_\gamma = \frac{\partial^2}{\partial \eta_\gamma \partial \eta_\gamma^T} \mathrm{KL}_{\mathrm{glob}}(\hat{\eta}_\gamma, 0). \tag{35}$$

In practice, we evaluate $\hat{H}_\gamma$ using automatic differentiation and Eq. 35 rather than the Schur complement.

D.3 Expressing g using global parameters only

Given a posterior quantity g, we again take advantage of the fact that the optimal local parameters can be found in closed form given global parameters. In general, g will be a function of the entire vector of variational parameters. However, in the same way that $\mathrm{KL}_{\mathrm{glob}}$ implicitly sets the local parameters at their optimum and is a function of only global parameters and the prior parameter t, we can construct an analogous mapping for g,

$$(t, \eta_\gamma) \mapsto g\Big(\big(\eta_\gamma, \hat{\eta}_z(\eta_\gamma, t)\big)\Big). \tag{36}$$

We illustrate this mapping when our quantity of interest is the in-sample expected posterior number of clusters.

Example 7. Let $g_{\mathrm{cl}}(\eta)$ denote our variational approximation to $\mathop{\mathbb{E}}_{\mathcal{P}(z|x)}[G_{\mathrm{cl}}(z)]$. Using the fact that $\mathcal{P}(z_n|\beta,\nu,x) = \mathcal{Q}(z_n|\hat{\eta}_{z_n})$ is available in closed form, we can then take

$$\begin{aligned}
g_{\mathrm{cl}}(\hat{\eta}) &:= \mathop{\mathbb{E}}_{\mathcal{Q}(\beta,\nu|\hat{\eta})}\left[\mathop{\mathbb{E}}_{\mathcal{P}(z|\beta,\nu,x)}[G_{\mathrm{cl}}(z)]\right] \\
&\approx \mathop{\mathbb{E}}_{\mathcal{P}(\beta,\nu|x)}\left[\mathop{\mathbb{E}}_{\mathcal{P}(z|\beta,\nu,x)}[G_{\mathrm{cl}}(z)]\right] = \mathop{\mathbb{E}}_{\mathcal{P}(z|x)}[G_{\mathrm{cl}}(z)] \Rightarrow \\
g_{\mathrm{cl}}(\eta) &= \sum_{k=1}^{K_{\max}-1}\left(1 - \prod_{n=1}^{N}\left(1 - \mathop{\mathbb{E}}_{\mathcal{Q}(\beta,\nu|\eta_\beta,\eta_\nu)}\left[\mathop{\mathbb{E}}_{\mathcal{P}(z_n|\beta,\nu,x)}[z_{nk}]\right]\right)\right).
\end{aligned}$$

In this way, $g_{\mathrm{cl}}(\eta)$ depends only on η_β and η_ν, which are much lower-dimensional than η_z, and retains nonlinearities in the map

$$\eta_\beta, \eta_\nu \mapsto \mathop{\mathbb{E}}_{\mathcal{Q}(\beta,\nu|\eta_\beta,\eta_\nu)}\left[\mathop{\mathbb{E}}_{\mathcal{P}(z_n|\beta,\nu,x)}[z_{nk}]\right].$$

△

The mapping Eq. 36 can be constructed for any posterior quantity g. Therefore, linearizing the global parameters using Eqs. 19 and 20 is sufficient: we do not need to invert the full Hessian and linearize the entire set of variational parameters, global and local.

D.4 Evaluating stick expectations

We describe how to compute expectations with repsect to the stick-breaking proportion ν_k. Let $f : \mathbb{R} \mapsto \mathbb{R}$ be a smooth function, and we are interested in expectations of the form

$$\mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta)}\left[f(\nu_k)\right].$$

For example, f might be $f(\nu_k) = \log \mathcal{P}(\nu_k)$, whose expectation appears in the KL divergence.

Recall that we chose the distribution on the logit-transformed stick-breaking proportions $\tilde{\nu}_k$ to be normally distributed. Let η_k^μ and η_k^σ be the location and scale, respectively, of the Gaussian distribution on $\tilde{\nu}_k$. Also let s be the sigmoid function, so that $\nu_k = s(\tilde{\nu}_k)$.

To compute expectations of a smooth function $f(\nu_k)$, the law of the unconscious statistician states that

$$\mathop{\mathbb{E}}_{\mathcal{Q}(\nu_k|\eta)}\left[f(\nu_k)\right] = \mathop{\mathbb{E}}_{\mathcal{Q}(\tilde{\nu}_k|\eta)}\left[f \circ s\left(\tilde{\nu}_k\right)\right].$$

By choosing $\mathcal{Q}(\tilde{\nu}_k|\eta)$ to be Gaussian, the right-hand side of is a Gaussian integral, which we approximate using GH quadrature with N_{GH} knots, located at ξ_g, weighted by ω_g:

$$\mathop{\mathbb{E}}_{\mathcal{Q}(\tilde{\nu}_k|\eta)}\left[f \circ s\left(\tilde{\nu}_k\right)\right] \approx \sum_{g=1}^{N_{\text{GH}}} \omega_g f \circ s\left(\eta_k^\sigma \xi_g + \eta_k^\mu\right) \tag{37}$$

Using GH quadrature to approximate the expectation is similar to the "reparameterization trick," only using GH points rather than standard normal draws.

D.5 Unconstrained variational parameterizations

Recall from Theorem 4 that we require the optimal variational parameters $\hat{\eta}$ to be in the interior of its domain. One way to achieve this is to use only *unconstrained* parameterizations for the component distributions of $\mathcal{Q}$. One such parameterization was presented in Example 6, where we let η_{z_n}, which parameterize the cluster assignments, be allowed to take any value in $\mathbb{R}^{K_{\max}-1}$; the assignment probabilities $m_n \in \mathbb{R}^{K_{\max}}$, which are constrained to sum to one, are then formed with an appropriate transform of the unconstrained parameters η_{z_n}

Other variables require careful parameterization as well. For instance, instead of parameterizing the normal distribution on logit-sticks $\tilde{\nu}_k$ using a mean and

variance, we let $\eta_{\nu_k} \in \mathbb{R}^2$ be the mean and *log* variance. The variance is constrained to be positive; the log-variance is unconstrained on the real line. In general, a real-valued parameter μ_i which must be constrained $a < \mu_i < b$ can be transformed to its unconstrained parameterization by letting

$$\eta_i = \log(\mu_i - a) - \log(b - \mu_i).$$

In the variational approximation to the GMM model, we let the component variables β_k be Normal-Wishart. In this case, the scale matrix of the Normal-Wishart, $W_k \in \mathbb{R}^{d \times d}$, is constrained be positive definite. Because W is symmetric, we only neeed $d(d+1)/2$ parameters to represent it. To form an unconstrained parameterization, we factorize W using the Cholesky decomposition,

$$W = L^T L,$$

where L is a lower-triangular matrix, with positive diagonal entries. The unconstrained parameterization of W is then taken to be the strictly lower-diagonal entries of L, along with the log of the diagonal entries of L.

E Additional Experimental Details

First, we review the Beta prior on the stick-breaking proportions, which are common to each model we considered. Then, we give some additional modeling details for each experiment.

E.1 The Beta prior

In the iris experiment, we considered $\alpha \in [0.1, 4.0]$. Over this range, the shape of the Beta $(1, \alpha)$ stick-breaking density varies considerably, as shown in Figure 12.

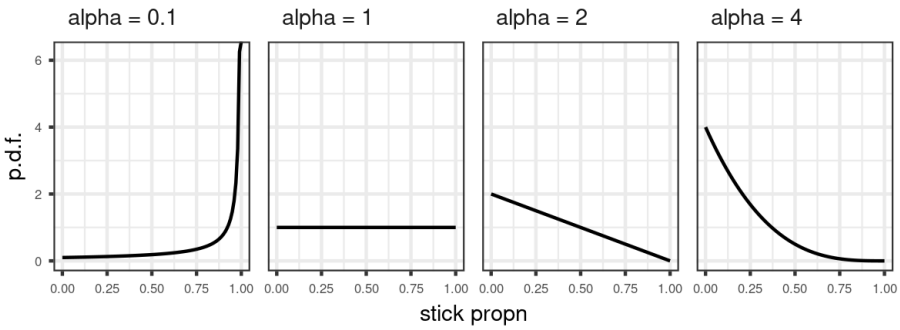


Figure 12: Probability density functions of Beta$(1, \alpha)$ distributions, under various α considered for the iris data set.

To help us understand the effect of the concentration parameter α we often use the following fact. Under the GEM(α) prior, the *a priori* expected number of distinct clusters in a dataset of size N is given by

$$\mathop{\mathbb{E}}_{\mathcal{P}(z|\pi)\mathcal{P}(\pi|\alpha)}[G_{\text{cl}}(z)] = \sum_{n=1}^{N} \frac{\alpha}{\alpha + n - 1}. \tag{38}$$

See Blackwell and MacQueen and Teh [2010, Equation 11].

E.2 Gaussian mixture modeling on iris data

The observations are vectors $x_n \in \mathbb{R}^d$, and we model each component with a multivariate Gaussian. In this model, $\beta_k = (\mu_k, \Lambda_k)$, where $\mu_k \in \mathbb{R}^d$, Λ_k is a $d \times d$ positive definite information matrix, and

$$\begin{aligned}
\mathcal{P}(x_n|\beta_k) &= \mathcal{N}\left(x_n|\mu_k, \Lambda_k^{-1}\right) \\
\log \mathcal{P}(x_n|\beta_k) &= -\frac{1}{2}(x_n - \mu_k)^T \Lambda_k (x_n - \mu_k) + \frac{1}{2}\log|\Lambda_k| + C. \\
&\quad (C \text{ does not depend on } \beta_k)
\end{aligned}$$

We let $\mathcal{P}_{\text{base}}(\beta_k)$ be the conjugate prior, which in this case is normal-Wishart:

$$\begin{aligned}
\mathcal{P}_{\text{base}}(\beta_k) &= \mathcal{NW}\left(\beta_k|\tau_0, n_0, p_0, V_0\right) \\
\log \mathcal{P}_{\text{base}}(\beta_k) &= -\frac{\tau_0}{2}(\mu_k - \mu_0)^T \Lambda_k (\mu_k - \mu_0) \\
&\quad + \frac{n_0 - p_0 - 1}{2}\log|\Lambda_k| - \frac{1}{2}\text{Tr}(V_0\Lambda_k) + C,
\end{aligned}$$

where (τ_0, n_0, p_0, V_0) are fixed prior parameters.

In this model, the conditionally conjugate variational distribution on β_k is normal-Wishart, which we denote as $\mathcal{Q}(\beta_k|\eta) = \mathcal{NW}\left(\beta_k|\eta_{\beta_k}\right)$, with η_{β_k} the normal-Wishart parameters. The conditionally conjugate variational distribution on z are multinomial.

Figure 13 shows the inferred clustering for $\alpha_0 = 2$, which recovers that there are three iris species.

E.3 Regression mixture modeling

The data. The data come from a publicly available data set of mice gene expression [Shoemaker et al., 2015]. Our analysis focuses on mice treated with the "A/California/04/2009" strain. We normalize the data as described in Shoemaker et al. [2015] and then apply the differential analysis tool EDGE [Storey et al., 2005] to rank the genes from most to least significantly differentially expressed. We run our analysis on the top $N = 1000$ genes.

The left plot of Figure 14 shows the measurements of a single gene over time. We model each gene as belonging to a latent component, where each component

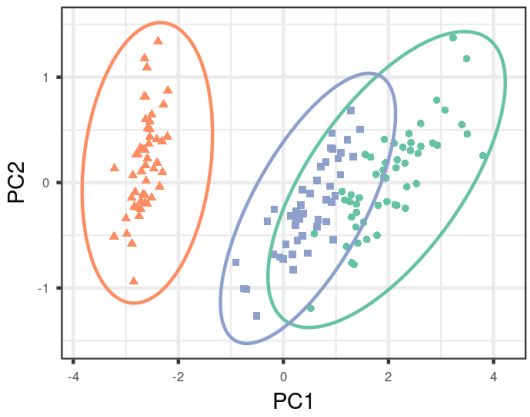


Figure 13: The iris data in principal component space and GMM fit at $\alpha = 2$. Colors denote inferred memberships and ellipses represent estimated covariances.

defines a smooth expression curve over time. Then, observations are drawn by adding i.i.d. noise to the smoothed curve along with a gene-specific offset.

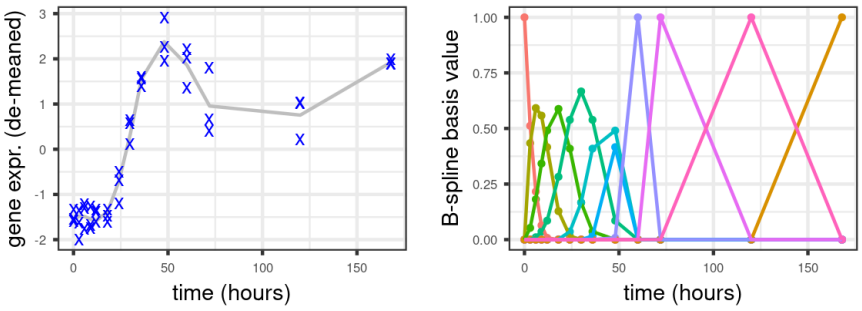


Figure 14: (Left) An example gene and its expression measured at 14 unique time points with three biological replicates at each time point. (Right) The cubic B-spline basis with 7 degrees of freedom, along with three indicator functions for the last three time points, $T = 72, 120, 168$.

The B-spline basis. Notice from Figure 14, which shows an example time-course for a single gene, that the time points are unevenly spaced, with more frequent observations at the beginning. Following Luan and Li [2003] we use cubic B-splines to smooth the time course expression data. Specifically, we model the first 11 time points using cubic B-splines with 7 degrees of freedom. For the last three time

points, $T = 72, 120, 168$ hours, we use indicator functions. That is, if $\tilde{A}$ is the design matrix where each column is a B-spline basis vector evaluated at the M measurement times, we append to $\tilde{A}$ three additional columns: in these columns, entries are 1 if $T = 72, 120$, or 168, receptively, and 0 otherwise. The resulting matrix is the full design matrix A. We use indicators for the last three time points for numerical stability; without the indicator columns, the matrix $\tilde{A}^T\tilde{A}$ is nearly singular because the later time points are more spread out. The left column of Figure 14 shows our basis functions.

The generative model. Eq. 21 gives the per-component conditional likelihood. We use a normal prior for the shifts b_n, a multivariate normal prior for the coefficients μ_k, and a gamma prior for the inverse variance τ_k. The prior on the mixture weights π are constructed using the stick-breaking construction in the main-text, and the cluster assignments z_n are drawn from a multinomial with wieghts π, as usual.

The variational approximation. The variational approximation, factorizes as

$$\mathcal{Q}(\zeta|\eta) = \left(\prod_{k=1}^{K_{\max}-1} \mathcal{Q}(\nu_k|\eta)\right)\left(\prod_{k=1}^{K_{\max}} \mathcal{Q}(\beta_k|\eta)\right)\left(\prod_{n=1}^{N} \mathcal{Q}(z_n|\eta)\mathcal{Q}(b_n|z_n,\eta)\right).$$

Note that the variational distribution for b_n conditions on z. We set $\mathcal{Q}(b_n|z_n = k, \eta)$ to be Gaussian with variational parameters dependent on k. For simplicity in this application, we let $\mathcal{Q}(\beta_k|\eta) = \delta(\beta_k|\eta)$, where $\delta(\cdot|\eta)$ denotes a point mass at a parameterized location.

As discussed in Example 6, the optimal distribution $\mathcal{Q}(z_n|\eta)$ is multinomial whose parameters can be set in closed form as a function of the global variational parameters only. We allow the distribution of b_n to depend on z_{nk} so that the its optimal distribution can also be set in closed form as a function of global parameters.

The optimal distribution $q(b_n|z_{nk} = 1, \eta)$ is Gaussian,

$$q(b_n|z_{nk} = 1, \eta) = \mathcal{N}\left(b_n|\hat{\mu}_{b_{nk}}, \hat{\sigma}^2_{b_{nk}}\right).$$

To define the optimal parameters $\hat{\mu}_{b_{nk}}, \hat{\sigma}^2_{b_{nk}}$, let

$$\begin{aligned}
\rho^{(1)}_{nk} &= \mathop{\mathbb{E}}_{\mathcal{Q}(\beta_k|\eta)}\left[\sum_{m=1}^{M} \tau_k(x_{nm} - A_m\mu_k)\right] + \tau_0\mu_0 \\
\rho^{(2)}_{nk} &= M \mathop{\mathbb{E}}_{\mathcal{Q}(\beta_k|\eta)}[\tau_k] + \tau_0,
\end{aligned}$$

where μ_0 and τ_0 are the prior mean and information on b_n, respectively.

The optimal parameters for the Gaussian distribution on b_n are given by

$$\begin{aligned}
\hat{\mu}_{b_{nk}} &= \rho^{(1)}_{nk}/\rho^{(2)}_{nk} \\
\hat{\sigma}^2_{b_{nk}} &= 1/\rho^{(2)}_{nk}.
\end{aligned}$$

Figure 15 shows the inferred smoothers $A\mathop{\mathbb{E}}_{\mathcal{Q}}[\mu_k]$ for selected clusters.

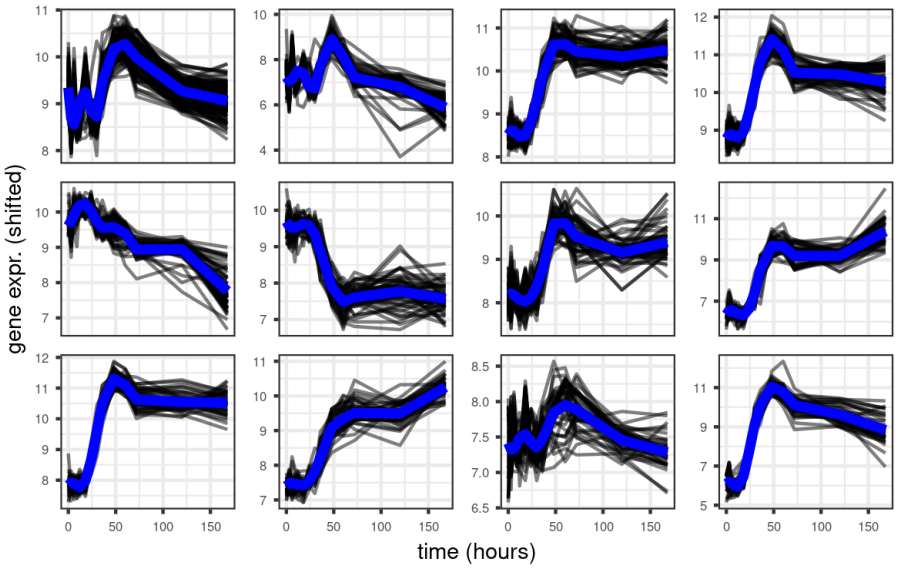


Figure 15: Inferred clusters in the mice gene expression dataset. Shown are the twelve most occupied clusters. In blue, the inferred cluster centroid. In grey, gene expressions averaged over replicates and shifted by their inferred intercepts.

E.4 fastSTRUCTURE

The generative process was described in the main text (Section 7.3). We detail here the variational approximation. Like in all our examples, the variational distribution is mean-field:

$$\mathcal{Q}(\zeta|\eta) = \left(\prod_{n=1}^{N}\prod_{k=1}^{K_{\max}-1}\mathcal{Q}(\nu_{nk}|\eta)\right)\left(\prod_{k=1}^{K_{\max}}\prod_{l=1}^{L}\mathcal{Q}(\beta_{kl}|\eta)\right)\left(\prod_{n=1}^{N}\prod_{l=1}^{L}\prod_{i=1}^{2}\mathcal{Q}(z_{nli}|\eta)\right).$$

We let all distributions be conditionally conjugate except for the sticks, which are logit-normal. Each membership indicator z_{nli} is categorical, and the allele frequencies β_{kl} are Dirichlet distributed.

In this model, we still call (β, ν) the global latent variables, even though they scale with the number of individuals N; they do not, however, scale with both the number of individuals and the number of loci like z does. Thus, we call z the local latent variables. The local variational parameters η_z can be set optimally in an analagous way as Example 6, except with the indices nk replaced with $nlik$.

The posterior quantities of interest in this application are the admixtures π_n. Figure 16 plots the inferred admixtures $\mathop{\mathbb{E}}_{\mathcal{Q}(\pi_n|\hat{\eta})}[\pi_n]$ for all individuals n.

In the approximate posterior with $\alpha_0 = 3$, there appear to be three dominant latent populations, which we arbitrarily label as populations 1, 2, and 3 (Figure 16).

The inferred admixture proportions generally correspond with geographic regions: Mbololo individuals are primarily population 1; Ngangao individuals are primarily population 2; and Chawia individuals are a mixture of populations 1, 2, and 3.

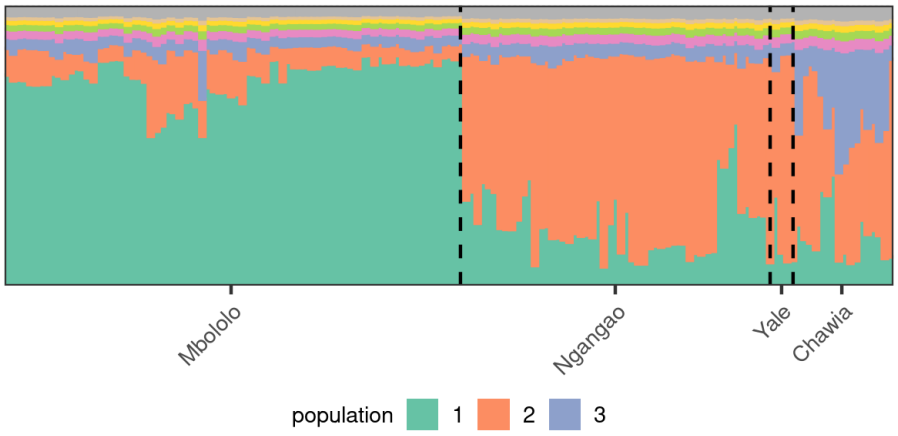


Figure 16: The inferred individual admixtures at $\alpha_0 = 3$. Each vertical strip is an individual and each color a latent population. Lengths of colored segments represent the inferred admixture proportions. Individuals are ordered by the geographic region from which they were sampled (Mbololo, Ngangao, Yale, and Chawia). In the text, we refer to the green, orange, and purple latent populations as population 1, 2, and 3, respectively.

F fastSTRUCTURE Supplemental Results

F.1 The expected number of populations

One posterior quantity of interest is the expected number of in-sample populations. Define

$$g_{\mathrm{cl},\tau}(\eta) = \mathop{\mathbb{E}}_{\mathcal{Q}(z|\eta)} \left[\sum_{k=1}^{K_{\max}} \mathbb{I}\left(\left(\sum_{n=1}^{N} \sum_{l=1}^{L} \sum_{i=1}^{2} z_{nlik} \right) > \tau \right) \right],$$

which is the expected number of populations in the data set that contains at least τ loci. We allow the option of setting $\tau > 0$ in order to count only the populations that comprise a non-negligible fraction of the data set.

The expected number of latent populations is sensitive to α (Figure 17). Without any thresholding ($\tau = 0$), the expected number of populations quickly increases as α increases; in fact, it nearly saturates at $K_{\max} = 20$ when $\alpha = 7$. This sensitivity is due to the fact that the non-thresholded quantity is highly dependent on the behavior of small, nearly unoccupied populations; even though the probability

of a single locus belonging to these rare populations is small, the probability that *none* of the $N \times L \times 2$ observed genotypes belong to these rare populations is non-negligible.

This motivates the use of thresholding in reporting the number of populations. We consider two thresholds, $\tau = 20$ and $\tau = 40$, corresponding to approximately 2% and 4% of the total number of loci in the data set, respectively. The thresholded estimates for the number of populations is still moderately sensitive to the value of α. When refitting the variational approximation at $\alpha = 0.5, 1, \ldots, 7$, the thresholded quantities vary between two and four latent populations.

The linearized variational parameters $\hat{\eta}_\gamma^{\text{lin}}(t)$ imperfectly captures the results observed by refitting. The linearized parameters and the refitted parameters almost perfectly agree on values of $g_{\text{cl},\tau}$ with $\tau = 0$. However, when $\tau = 20$, the linearized parameters underestimated the true sensitivity of $g_{\text{cl},\tau}$ found by refitting. In particular, the linearized parameters failed to produce the reduction to two latent populations at $\alpha = 0.5$ observed in the refits.

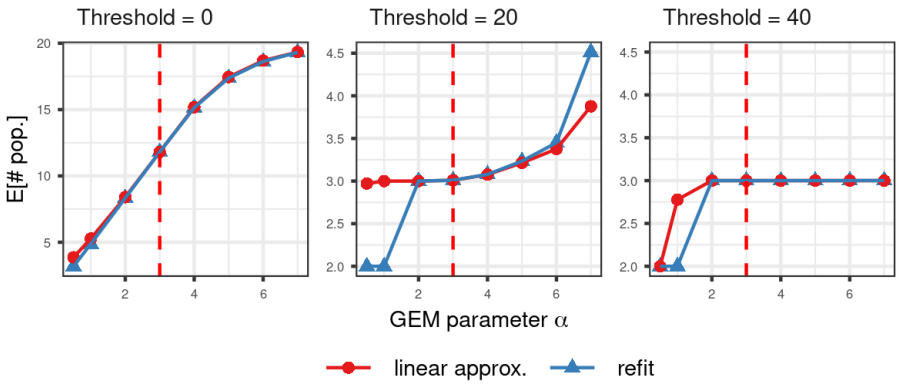


Figure 17: The expected number of (thresholded) populations in the thrush data as α varies. We computed the linear approximation at $\alpha_0 = 3$, and we compare the results under the linearly approximated variational parameters with the results observed after refitting. Thresholds at $\tau = 20$ and $\tau = 40$ corresponding to approximately 2% and 4% of the total number of loci in the data set, respectively.

We provide some more intuition concerning the thresholded estimate for the number of populations. The posterior quantity $g_{\text{cl},\tau}$ is closely related to the expected number of loci belonging to each population, defined as

$$g_{\text{loci}}(\eta; k) = \mathop{\mathbb{E}}_{\mathcal{Q}(z|\eta)} \left[\sum_{n=1}^{N} \sum_{l=1}^{L} \sum_{i=1}^{2} z_{nlik} \right].$$

Figure 18 plots g_{loci} for the first six populations as α varies. The expected number of loci at the initial fit, $g_{\text{loci}}(\hat{\eta}(\alpha_0); k)$, is at least 100 for populations

$k = 1, 2$, and 3 and less than 15 for the remaining populations. A sample of assignments $z \sim \mathcal{Q}(z|\hat{\eta}(\alpha_0))$ will almost always have at least τ loci allocated to populations 1, 2, and 3, while the allocations to each remaining population will almost always be below τ, for either $\tau = 20$ or $\tau = 40$. Thus, at $\alpha = \alpha_0$ there then are clearly 3 populations by our definition of $g_{\text{cl},\tau}$, for either τ.

At $\alpha = 7$, the expected number of loci belonging to population 4 increases to approximately 20, and a new population emerges above the threshold at $\tau = 20$. Both the linearized and the refitted variational parameters agree on this shift in allocation to population 4. On the other hand, under the refitted variational parameters at $\alpha = 0.5$, the expected number of loci belonging to population 3 decreases to seven, below the threshold $\tau = 20$. Thus, the expected number of latent populations with allocations above the threshold $\tau = 20$ decreases to two. The linearized parameters under-estimated this decrease in allocation to population 3, and therefore continued to estimate three latent populations even at $\alpha = 0.5$.

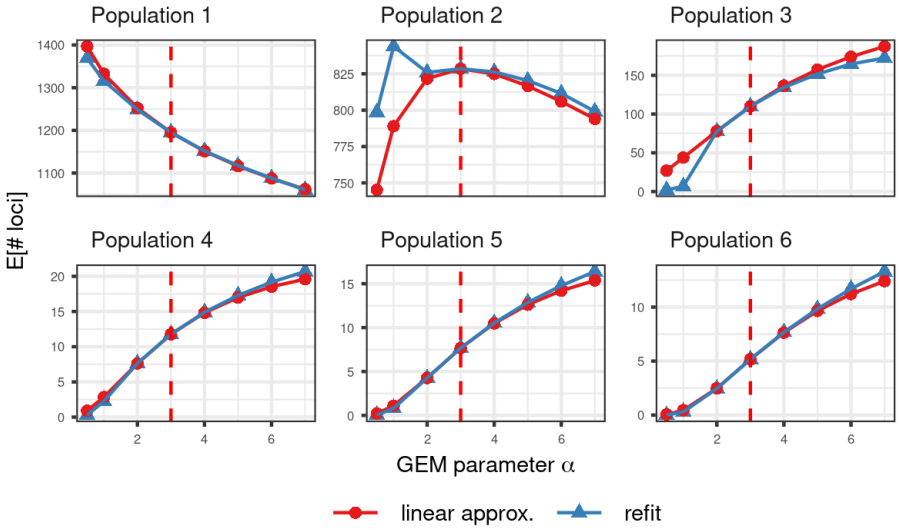


Figure 18: The expected number of loci per population as α varies.

F.2 Limitations of local sensitivity

Recall from Section 7.3 and Figure 10 that the linear approximation failed to capture the change in the admixture proportion of an individual, $n = 25$ after a worst-case functional perturbation.

Figure 19 examines individual $n = 25$ more closely. The bottom row plots this individual's admixture proportions as t varies from 0 to 1 in the perturbed prior $\mathcal{P}(\nu|t) = \mathcal{P}_0(\nu_k) \exp(t\phi_{\text{wc}}(\nu_k))$. The linearized parameters poorly captured

the change in admixture proportions observed after refitting, particularly for populations 1 and 2, for values of t close to 1. Even though we retain non-linearities in the mapping from variational parameters to the posterior statistic, for this perturbation, the mapping from prior parameter t to the relevant variational parameters is highly non-linear. This latter mapping is what we linearize and what causes our approximation to fail in this case. Specifically, the variational location parameter on the first stick-breaking proportion is concave as a function of t – the location parameter increases for small t, then decreases as $t \to 1$. However, $\hat{\eta}^{\text{lin}}(t)$ linearizes the relationship between the location parameter and t. Therefore, the corresponding admixture mixture proportion of population 1 is over-estimated under the linearized variational parameters. Furthermore, because our linearized variational parameters over-estimated the length of the first stick, and the second admixture proportion is a product of the remaining stick times the second stick-breaking proportion, the linearized variational parameters then under-estimates the admixture proportion of population 2.

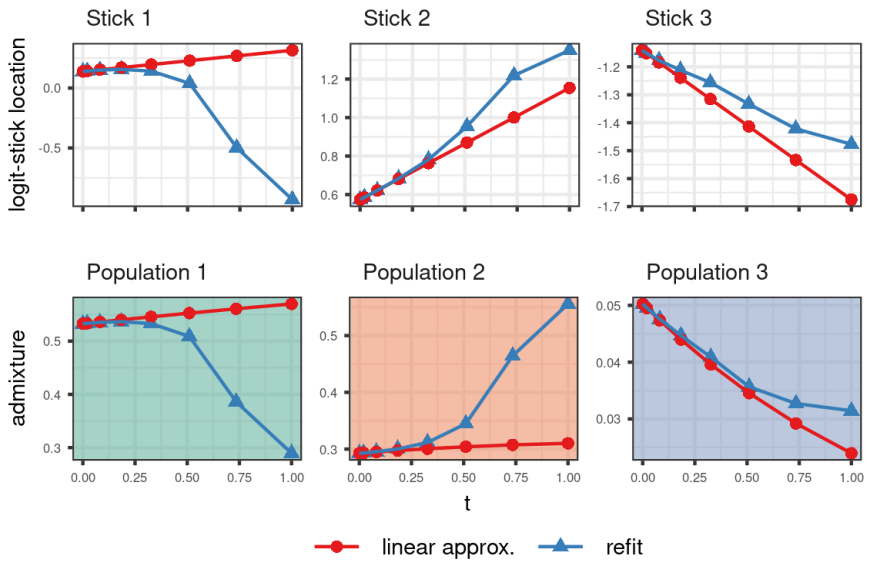


Figure 19: An individual ($n = 26$) for which the linearly approximated variational parameters poorly captured the change in admixture observed after refitting as $t \to 1$. (Top row) the change in location parameter of the normally distributed logit-sticks, for the first three sticks. The response here is a variational parameter, so the approximation (red) is necessarily linear with respect to t. (Bottom row) the change in the inferred admixtures for populations 1, 2, and 3.

Figure 20 shows a similar situation for individual $n = 74$. The linearized variational parameters grossly over-estimated the length of the first stick, resulting in the later admixture proportions being under-estimated. The third admixture proportion was particularly poorly approximated under the linearized variational parameters. Given the recursive nature of the relationship between admixtures and stick-breaking proportions, errors at early sticks affect later admixture proportions. Fully linearizing the mapping $t \mapsto g(\hat{\eta}(t))$ to form the approximation $g^{\text{lin}}(t)$ avoids this problem. In this example, $g^{\text{lin}}(t)$ outperforms $g(\hat{\eta}^{\text{lin}}(t))$, with g being the admixture proportion of population 3. In our experience, computing $g(\hat{\eta}^{\text{lin}}(t))$, and thus retaining non-linearities in the mapping from $\eta \mapsto g(\eta)$, is usually beneficial to the quality of the approximation. It is likely that $g(\hat{\eta}^{\text{lin}}(t))$ outperforms $g^{\text{lin}}(t)$ for most posterior quantities, though as we see in Figure 20, this is not guaranteed to always be true.

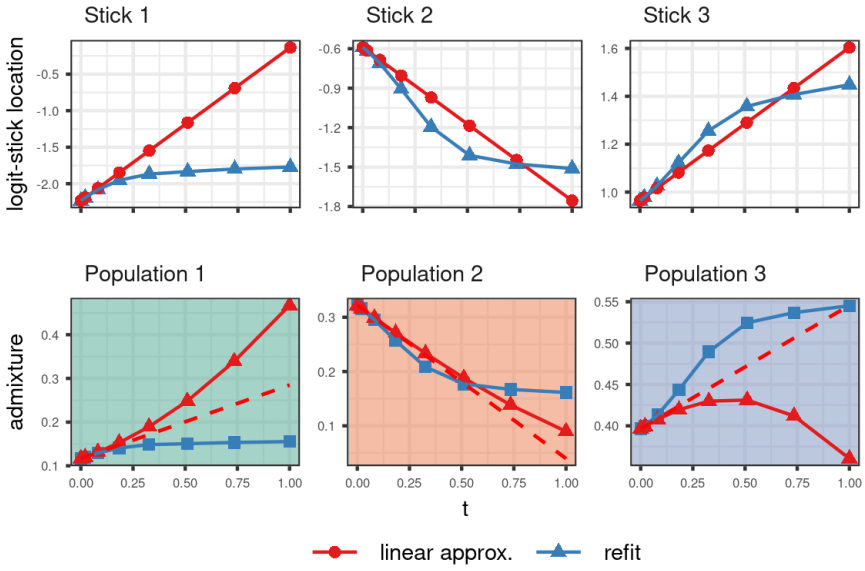


Figure 20: An example where linearizing the posterior quantity itself outperforms linearizing the variational parameters only. Shown are logit-stick location parameters (top row) and inferred admixtures (bottom row) for individual $n = 74$ and populations $k = 1, 2$ and 3. Dashed red is the approximation $g^{\text{lin}}(t)$ formed by linearizing the inferred admixture $\underset{\mathcal{Q}}{\mathbb{E}}[\pi_{nk}]$ with respect to prior parameter t. On the admixture proportion of population 3, $g^{\text{lin}}(t)$ outperforms $g(\hat{\eta}^{\text{lin}}(t))$ (solid red).

A Swiss Army Infinitesimal Jackknife

Ryan Giordano
rgiordano@berkeley.edu

Will Stephenson
wtstephe@mit.edu

Runjing Liu
runjing_liu@berkeley.edu

Michael I. Jordan
jordan@cs.berkeley.edu

Tamara Broderick
tbroderick@csail.mit.edu

February 7, 2020

Abstract

The error or variability of machine learning algorithms is often assessed by repeatedly re-fitting a model with different weighted versions of the observed data. The ubiquitous tools of cross-validation (CV) and the bootstrap are examples of this technique. These methods are powerful in large part due to their model agnosticism but can be slow to run on modern, large data sets due to the need to repeatedly re-fit the model. In this work, we use a linear approximation to the dependence of the fitting procedure on the weights, producing results that can be faster than repeated re-fitting by an order of magnitude. This linear approximation is sometimes known as the "infinitesimal jackknife" in the statistics literature, where it is mostly used as a theoretical tool to prove asymptotic results. We provide explicit finite-sample error bounds for the infinitesimal jackknife in terms of a small number of simple, verifiable assumptions. Our results apply whether the weights and data are stochastic or deterministic, and so can be used as a tool for proving the accuracy of the infinitesimal jackknife on a wide variety of problems. As a corollary, we state mild regularity conditions under which our approximation consistently estimates true leave-k-out cross-validation for any fixed k. These theoretical results, together with modern automatic differentiation software, support the application of the infinitesimal jackknife to a wide variety of practical problems in machine learning, providing a "Swiss Army infinitesimal jackknife." We demonstrate the accuracy of our methods on a range of simulated and real datasets.

1 Introduction

Statistical machine learning methods are increasingly deployed in real-world problem domains where they are the basis of decisions affecting individuals'

employment, savings, health, and safety. Unavoidable randomness in data collection necessitates understanding how our estimates, and resulting decisions, might have differed had we observed different data. Both cross validation (CV) and the bootstrap attempt to diagnose this variation and are widely used in classical data analysis. But these methods are often prohibitively slow for modern, massive datasets, as they require running a learning algorithm on many slightly different datasets. In this work, we propose to replace these many runs with a single perturbative approximation. We show that the computation of this approximation is far cheaper than the classical methods, and we provide theoretical conditions that establish its accuracy.

Many data analyses proceed by minimizing a loss function of exchangeable data. Examples include empirical loss minimization and M-estimation based on product likelihoods. Since we typically do not know the true distribution generating the data, it is common to approximate the dependence of our estimator on the data via the dependence of the estimator on the empirical distribution. In particular, we often form a new, proxy dataset using random or deterministic modifications of the empirical distribution, such as randomly removing k datapoints for leave-k-out CV. A proxy dataset obtained in this way can be represented as a weighting of the original data. From a set of such proxy datasets we can obtain estimates of uncertainty, including estimates of bias, variance, and prediction accuracy.

As data and models grow, the cost of repeatedly solving a large optimization problem for a number of different values of weights can become impractically large. Conversely, though, larger datasets often exhibit greater regularity; in particular, under fairly general conditions, limit laws based on independence imply that an optimum exhibits diminishing dependence on any fixed set of data points. We use this observation to derive a linear approximation to resampling that needs to be calculated only once, but which nonetheless captures the variability inherent in the repeated computations of classical CV. Our method is an instance of the *infinitesimal jackknife* (IJ), a general methodology that was historically a precursor to cross-validation and the bootstrap [Jaeckel, 1972, Efron, 1982]. Part of our argument is that variants of the IJ should be reconsidered for modern large-scale applications because, for smooth optimization problems, the IJ can be calculated automatically with modern automatic differentiation tools [Baydin et al., 2017].

By using this linear approximation, we incur the cost of forming and inverting a matrix of second derivatives with size equal to the dimension of the parameter space, but we avoid the cost of repeatedly re-optimizing the objective. As we demonstrate empirically, this tradeoff can be extremely favorable in many problems of interest.

Our approach aims to provide a felicitous union of two schools of thought. In statistics, the IJ is typically used to prove normality or consistency of other estimators [Fernholz, 1983, Shao, 1993, Shao and Tu, 2012]. However, the conditions that are required for these asymptotic analyses to hold are prohibitively restrictive for machine learning—specifically, they require objectives with bounded gradients. A number of recent papers in machine learning have provided related

linear approximations for the special case of leave-one-out cross-validation [Koh and Liang, 2017, Rad and Maleki, 2018, Beirami et al., 2017], though their analyses lack the generality of the statistical perspective.

We combine these two approaches by modifying the proof of the Fréchet differentiability of M-estimators developed by Clarke [1983]. Specifically, we adapt the proof away from the question of Fréchet differentiability within the class of all empirical distributions to the narrower problem of approximating the exact re-weighting on a particular dataset with a potentially restricted set of weights. This limitation of what we expect from the approximation is crucial; it allows us to bound the error in terms of a complexity measure of the set of derivatives of the observed objective function, providing a basis for non-asymptotic applications in large-scale machine learning, even for objectives with unbounded derivatives. Together with modern automatic differentiation tools, these results extend the use of the IJ to a wider range of practical problems. Thus, our "Swiss Army infinitesimal jackknife," like the famous Swiss Army knife, is a single tool with many different functions.

2 Methods and Results

2.1 Problem definition

We consider the problem of estimating an unknown parameter $\theta \in \Omega_\theta \subseteq \mathbb{R}^D$, with a compact Ω_θ and a dataset of size N. Our analysis will proceed entirely in terms of a fixed dataset, though we will be careful to make assumptions that will plausibly hold for all N under suitably well-behaved random sampling. We define our estimate, $\hat{\theta} \in \Omega_\theta$, as the root of a weighted estimating equation. For each $n = 1, \ldots, N$, let $g_n(\theta)$ be a function from Ω_θ to $\mathbb{R}^D$. Let w_n be a real number, and let w be the vector collecting the w_n. Then $\hat{\theta}$ is defined as the quantity that satisfies

$$\hat{\theta}(w) := \quad \theta \text{ such that } \frac{1}{N}\sum_{n=1}^{N} w_n g_n(\theta) = 0. \tag{1}$$

We will impose assumptions below that imply at least local uniqueness of $\hat{\theta}(w)$; see the discussion following Assumption 2 in Section 2.3.

As an example, consider a family of continuously differentiable loss functions $f(\cdot, \theta)$ parameterized by θ and evaluated at data points $x_n, n = 1, \ldots, N$. If we want to solve the optimization problem $\hat{\theta} = \underset{\theta \in \Omega_\theta}{\operatorname{argmin}} \frac{1}{N}\sum_{n=1}^{N} f(x_n, \theta)$, then we take $g_n(\theta) = \partial f(x_n, \theta)/\partial\theta$ and $w_n \equiv 1$. By keeping our notation general, we will be able to analyze a more general class of problems, such as multi-stage optimization (see Section 6). However, to aid intuition, we will sometimes refer to the $g_n(\theta)$ as "gradients" and their derivatives as "Hessians."

When equation (1) is not degenerate (we articulate precise conditions below), $\hat{\theta}$ is a function of the weights through solving the estimating equation, and we

write $\hat{\theta}(w)$ to emphasize this. We will focus on the case where we have solved equation (1) for the weight vector of all ones, $1_w := (1, \ldots, 1)$, which we denote $\hat{\theta}_1 := \hat{\theta}(1_w)$.

A re-sampling scheme can be specified by choosing a set $W \subseteq \mathbb{R}^N$ of weight vectors. For example, to approximate leave-k-out CV, one repeatedly computes $\hat{\theta}(w)$ where w has k randomly chosen zeros and all ones otherwise. Define W_k as the set of every possible leave-k-out weight vector. Showing that our approximation is good for all leave-k-out analyses with probability one is equivalent to showing that the approximation is good for all $w \in W_k$.

In the case of the bootstrap, W contains a fixed number B of randomly chosen weight vectors, $w_b^* \overset{iid}{\sim} \text{Multinomial}\left(N, N^{-1}\right)$ for $b = 1, \ldots, B$, so that $\sum_{n=1}^N w_{bn}^* = N$ for each b. Note that while w_n or w_{bn}^* are scalars, w_b^* is a vector of length N. The distribution of $\hat{\theta}(w_b^*) - \hat{\theta}(1_w)$ is then used to estimate the sampling variation of $\hat{\theta}_1$. Define this set $W_B^* = \{w_1^*, \ldots, w_B^*\}$. Note that W_B^* is stochastic and is a subset of all weight vectors that sum to N.

In general, W can be deterministic or stochastic, may contain integer or non-integer values, and may be determined independently of the data or jointly with it. As with the data, our results hold for a given W, but in a way that will allow natural high-probability extensions to stochastic W.

2.2 Linear approximation

The main problem we solve is the computational expense involved in evaluating $\hat{\theta}(w)$ for all the $w \in W$. Our contribution is to use only quantities calculated from $\hat{\theta}_1$ to approximate $\hat{\theta}(w)$ for all $w \in W$, without re-solving equation (1). Our approximation is based on the derivative $\frac{d\hat{\theta}(w)}{dw^T}$, whose existence depends on the derivatives of $g_n(\theta)$, which we assume to exist, and which we denote as $h_n(\theta) := \frac{\partial g_n(\theta)}{\partial \theta^T}$. We use this notation because $h_n(\theta)$ would be the Hessian of a term of the objective in the case of an optimization problem. We make the following definition for brevity.

Definition 1. The fixed point equation and its derivative are given respectively by

$$G(\theta, w) := \frac{1}{N} \sum_{n=1}^N w_n g_n(\theta)$$
$$H(\theta, w) := \frac{1}{N} \sum_{n=1}^N w_n h_n(\theta).$$

Note that $G\left(\hat{\theta}(w), w\right) = 0$ because $\hat{\theta}(w)$ solves equation (1) for w. We define $H_1 := H\left(\hat{\theta}_1, 1_w\right)$ and define the weight difference as $\Delta w = w - 1_w \in \mathbb{R}^N$. When H_1 is invertible, one can use the implicit function theorem and the chain rule to

show that the derivative of $\hat{\theta}(w)$ with respect to w is given by

$$\begin{aligned}\frac{d\hat{\theta}(w)}{dw^T}|_{1_w}\Delta w &= -H_1^{-1}\frac{1}{N}\sum_{n=1}^{N} g_n\left(\hat{\theta}_1\right)\Delta w \\ &= -H_1^{-1}G\left(\hat{\theta}_1, \Delta w\right).\end{aligned}$$

This derivative allows us to form a first-order approximation to $\hat{\theta}(w)$ at $\hat{\theta}_1$.

Definition 2. Our linear approximation to $\hat{\theta}(w)$ is given by

$$\hat{\theta}_{\mathrm{IJ}}(w) := \hat{\theta}_1 - H_1^{-1}G\left(\hat{\theta}_1, \Delta w\right).$$

We use the subscript "IJ" for "infinitesimal jackknife," which is the name for this estimate in the statistics literature [Jaeckel, 1972, Shao, 1993]. Because $\hat{\theta}_{\mathrm{IJ}}$ depends only on $\hat{\theta}_1$ and Δw, and not on solutions at any other values of w, there is no need to re-solve equation (1). Instead, to calculate $\hat{\theta}_{\mathrm{IJ}}$ one must solve a linear system involving H_1. Recalling that θ is D-dimensional, the calculation of H_1^{-1} (or a factorization that supports efficient solution of linear systems) can be $O\left(D^3\right)$. However, once H_1^{-1} is calculated or H_1 is factorized, calculating our approximation $\hat{\theta}_{\mathrm{IJ}}(w)$ for each new weight costs only as much as a single matrix-vector multiplication. Furthermore, H_1 often has a sparse structure allowing H_1^{-1} to be calculated more efficiently than a worst-case scenario (see Section 6 for an example). In more high-dimensional examples with dense Hessian matrices, such as neural networks, one may need to turn to approximations such as stochastic second-order methods [Koh and Liang, 2017, Agarwal et al., 2017] and conjugate gradient [Wright and Nocedal, 1999]. Indeed, even in relatively small or sparse problems, the vast bulk of the computation required to calculate $\hat{\theta}_{\mathrm{IJ}}$ is in the computation of H_1^{-1}. We leave the important question of approximate calculation of H_1^{-1} for future work.

2.3 Assumptions and results

We now state our key assumptions and results, which are sufficient conditions under which $\hat{\theta}_{\mathrm{IJ}}(w)$ will be a good approximation to $\hat{\theta}(w)$. We defer most proofs to Appendix A. We use $\|\cdot\|_{op}$ to denote the matrix operator norm, $\|\cdot\|_2$ to denote the L_2 norm, and $\|\cdot\|_1$ to denote the L_1 norm. For quantities like g and h, which have dimensions $N \times D$ and $N \times D \times D$ respectively, we apply the L_p norm to the vectorized version of arrays. For example, $\frac{1}{\sqrt{N}}\|h(\theta)\|_2 = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{D}\sum_{j=1}^{D}\left[h_n(\theta)\right]_{ij}^2}$ which is the square root of a sample average over $n \in [N]$.

We state all assumptions and results for a fixed N, a given estimating equation vector $g(\theta)$, and a fixed class of weights W. Although our analysis proceeds with these quantities fixed, we are careful to make only assumptions that can plausibly hold for all N and/or for randomly chosen W under appropriate regularity conditions.

Assumption 1 (Smoothness)**.** *For all $\theta \in \Omega_\theta$, each $g_n(\theta)$ is continuously differentiable in θ.*

The smoothness in Assumption 1 is necessary for a local approximation like Definition 2 to have any hope of being useful.

Assumption 2 (Non-degeneracy)**.** *For all $\theta \in \Omega_\theta$, $H(\theta, 1_w)$ is non-singular, with $\sup_{\theta\in\Omega_\theta} \left\| H(\theta, 1_w)^{-1} \right\|_{op} \leq C_{op} < \infty$.*

Without Assumption 2, the derivative in Definition 2 would not exist. For an optimization problem, Definition 2 amounts to assuming that the Hessian is strongly positive definite, and, in general, assures that the solution $\hat{\theta}_1$ is unique. Under our assumptions, we will show later that, additionally, $\hat{\theta}(w)$ is unique in a neighborhood of $\hat{\theta}_1$; see Lemma 6 of Appendix A. Furthermore, by fixing C_{op}, if we want to apply Assumption 2 for $N \to \infty$, we will require that H_1 remains strongly positive definite.

Assumption 3 (Bounded averages)**.** *There exist finite constants C_g and C_h such that $\sup_{\theta\in\Omega_\theta} \frac{1}{\sqrt{N}} \|g(\theta)\|_2 \leq C_g < \infty$ and $\sup_{\theta\in\Omega_\theta} \frac{1}{\sqrt{N}} \|h(\theta)\|_2 \leq C_h < \infty$.*

Assumption 3 essentially states that the sample variances of the gradients and Hessians are uniformly bounded. Note that it does not require that these quantities are bounded term-wise. For example, we allow $\sup_n \|g_n(\theta)\|_2^2 \underset{N\to\infty}{\longrightarrow} \infty$, as long as $\sup_n \frac{1}{N} \|g_n(\theta)\|_2^2$ remains bounded. This is a key advantage of the present work over many past applications of the IJ to M-estimation, which require $\sup_n \|g_n(\theta)\|_2^2$ to be uniformly bounded for all N [Shao and Tu, 2012, Beirami et al., 2017].

In both machine learning and statistics, $\sup_n \|g_n(\theta)\|_2^2$ is rarely bounded, though $\frac{1}{N} \|g(\theta)\|_2^2$ often is. As a simple example, suppose that $\theta \in \mathbb{R}^1$, $x_n \sim \mathcal{N}(0,1)$, and $g_n = \theta - x_n$, as would arise from the squared error loss $f_n(x_n, \theta) = \frac{1}{2}(\theta - x_n)^2$. Fix a θ and let $N \to \infty$. Then $\sup_n \|g_n(\theta)\|_2^2 \to \infty$ because $\sup_n |x_n| \to \infty$, but $\frac{1}{N} \|g(\theta)\|_2^2 \to \theta^2 + 1$ by the law of large numbers.

Assumption 4 (Local smoothness)**.** *There exists a $\Delta_\theta > 0$ and a finite constant L_h such that, $\left\| \theta - \hat{\theta}_1 \right\|_2 \leq \Delta_\theta$ implies that $\frac{\left\| h(\theta) - h(\hat{\theta}_1) \right\|_2}{\sqrt{N}} \leq L_h \left\| \theta - \hat{\theta}_1 \right\|_2$.*

The constants defined in Assumption 4 are needed to calculate our error bounds explicitly.

Assumptions 1–4 are quite general and should be expected to hold for many reasonable problems, including holding uniformly asymptotically with high probability for many reasonable data-generating distributions, as the following lemma shows.

Lemma 1 (The assumptions hold under uniform convergence)**.** *Let Ω_θ be a compact set, and let $g_n(\theta)$ be twice continuously differentiable IID random functions for $n \in [N]$. (The function is random but θ is not—for example,*

$\mathbb{E}[g_n(\theta)]$ *is still a function of* θ*.) Define* $r_n(\theta) := \frac{\partial^2 g_n(\theta)}{\partial\theta\partial\theta}$*, so* $r_n(\theta)$ *is a* $D\times D\times D$ *tensor.*

Assume that we can exchange integration and differentiation, that $\mathbb{E}[h_n(\theta)]$ *is non-singular for all* $\theta \in \Omega_\theta$*, and that all of* $\mathbb{E}\left[\sup_{\theta\in\Omega_\theta} \|g_n(\theta)\|_2^2\right]$, $\mathbb{E}\left[\sup_{\theta\in\Omega_\theta} \|h_n(\theta)\|_2^2\right]$, *and* $\mathbb{E}\left[\sup_{\theta\in\Omega_\theta} \|r_n(\theta)\|_2^2\right]$ *are finite.*

Then $\lim_{N\to\infty} P(\textit{Assumptions 1–4 hold}) = 1$.

Lemma 1 follows from the uniform convergence results of Theorems 9.1 and 9.2 in Keener [2011]. See Appendix A.4 for a detailed proof. A common example to which Lemma 1 would apply is where x_n are well-behaved IID data and $g_n(\theta) = \gamma(x_n, \theta)$ for an appropriately smooth estimating function $\gamma(\cdot, \theta)$. See Keener [2011, Chapter 9] for more details and examples, including applications to maximum likelihood estimators on unbounded domains.

Assumptions 1–4 apply to the estimating equation. We also require a boundedness condition for W.

Assumption 5 (Bounded weight averages)**.** *The quantity* $\frac{1}{\sqrt{N}}\|w\|_2$ *is uniformly bounded for* $w \in W$ *by a finite constant* C_w.

Our final requirement is considerably more restrictive, and contains the essence of whether or not $\hat{\theta}_{\mathrm{IJ}}(w)$ will be a good approximation to $\hat{\theta}(w)$.

Condition 1 (Set complexity)**.** *There exists a* $\delta \geq 0$ *and a corresponding set* $W_\delta \subseteq W$ *such that*

$$\max_{w\in W_\delta} \sup_{\theta\in\Omega_\theta} \left\| \frac{1}{N}\sum_{n=1}^N (w_n - 1)\, g_n(\theta) \right\|_1 \leq \delta \quad \textit{and}$$
$$\max_{w\in W_\delta} \sup_{\theta\in\Omega_\theta} \left\| \frac{1}{N}\sum_{n=1}^N (w_n - 1)\, h_n(\theta) \right\|_1 \leq \delta.$$

Condition 1 is central to establishing when the approximation $\hat{\theta}_{\mathrm{IJ}}(w)$ is accurate. For a given δ, W_δ will be the class of weight vectors for which $\hat{\theta}_{\mathrm{IJ}}(w)$ is accurate to within order δ. Trivially, $1_w \in W_\delta$ for $\delta = 0$, so W_δ is always non-empty, even for arbitrarily small δ. The trick will be to choose a small δ that still admits a large class W_δ of weight vectors. In Section 3 we will discuss Condition 1 in more depth, but it will help to first state our main theorem.

Definition 3. The following constants are given by quantities in Assumptions 1–5 .

$$C_{\mathrm{IJ}} := 1 + DC_w L_h C_{op}$$
$$\Delta_\delta := \min\left\{ \Delta_\theta C_{op}^{-1}, \frac{1}{2} C_{\mathrm{IJ}}^{-1} C_{op}^{-1} \right\}.$$

Note that, although the parameter dimension D occurs explicitly only once in Definition 3, all of C_w, C_{op}, and L_h in general might also contain dimension

dependence. Additionally, the bound δ in Condition 1, a measure of the set complexity of the parameters, will typically depend on dimension. However, the particular place where the parameter dimension enters will depend on the problem and asymptotic regime, and our goal is to provide an adaptable toolkit for a wide variety of problems.

We are now ready to state our main result.

Theorem 1 (Error bound for the approximation)**.** *Under Assumptions 1–5 and Condition 1,*

$$\delta \leq \Delta_\delta \Rightarrow \max_{w \in W_\delta} \left\| \hat{\theta}_{IJ}(w) - \hat{\theta}(w) \right\|_2 \leq 2C_{op}^2 C_{IJ} \delta^2.$$

We stress that Theorem 1 bounds only the difference between $\hat{\theta}_{\mathrm{IJ}}(w)$ and $\hat{\theta}(w)$. Theorem 1 alone does not guarantee that $\hat{\theta}_{\mathrm{IJ}}(w)$ converges to any hypothetical infinite population quantity. We see this as a strength, not a weakness. To begin with, convergence to an infinite population requires stronger assumptions. Contrast, for example, the Fréchet differentiability work of Clarke [1983], on which our work is based, with the stricter requirements in the proof of consistency in Shao [1993]. Second, machine learning problems may not naturally admit a well-defined infinite population, and the dataset at hand may be of primary interest. Finally, by analyzing a particular sample rather than a hypothetical infinite population, we can bound the error in terms of the quantities C_{IJ} and Δ_δ, which can actually be calculated from the data at hand.

Still, Theorem 1 is useful to prove asymptotic results about the difference $\left\| \hat{\theta}_{\mathrm{IJ}}(w) - \hat{\theta}(w) \right\|_2$. As an illustration, we now show that the uniform consistency of leave-k-out CV follows from Theorem 1 by a straightforward application of Hölder's inequality.

Corollary 1 (Consistency for leave-k-out CV)**.** *Assume that Assumptions 1–5 hold uniformly for all N. Fix an integer k, and let*

$$W_k := \{w : w_n = 0 \text{ in } k \text{ entries and } 1 \text{ otherwise}\}.$$

Then, for all N, there exists a constant C_K such that

$$\begin{aligned} \sup_{w \in W_k} \left\| \hat{\theta}_{IJ}(w) - \hat{\theta}(w) \right\|_2 &\leq C_K \frac{\|g\|_\infty^2}{N^2} \\ &\leq C_K \frac{\max\{C_g, C_h\}^2}{N}. \end{aligned}$$

Proof. For $w \in W_k$, $\frac{\|\Delta w\|_2}{\sqrt{N}} = \sqrt{\frac{K}{N}}$. Define $C_{gh} := \max\{C_g, C_h\}$. By Assumption 3, $\|g\|_2 / \sqrt{N} \leq C_{gh}$ and $\|h\|_2 / \sqrt{N} \leq C_{gh}$ for all N. By Hölder's inequality,

$$\begin{aligned} &\sup_{w \in W} \sup_{\theta \in \Omega_\theta} \left\| \frac{1}{N} \sum_{n=1}^N (w_n - 1) g_n(\theta) \right\|_1 \\ &\leq \sup_{w \in W} \|w - 1_w\|_1 \sup_{\theta \in \Omega_\theta} \frac{\|g\|_\infty}{N} = K \frac{\|g\|_\infty}{N} \leq K \frac{C_{gh}}{\sqrt{N}}, \end{aligned}$$

with a similar bound for $\|h\|_2$. Consequently, for N large enough, Condition 1 is satisfied with $W_\delta = W_k$ and either $\delta = K\frac{\|g\|_\infty}{N}$ or $\delta = K\frac{C_{gh}}{\sqrt{N}}$. The result then follows from Theorem 1. □

3 Examples

The moral of Theorem 1 is that, under Assumptions 1–5 and Condition 1, $\left\|\hat{\theta}_{\mathrm{IJ}} - \hat{\theta}(w)\right\| = O\left(\delta^2\right)$ for $w \in W_\delta$. That is, if we can make δ small enough, W_δ big enough, and still satisfy Condition 1, then $\hat{\theta}_{\mathrm{IJ}}(w)$ is a good approximation to $\hat{\theta}(w)$ for "most" w, where "most" is defined as the size of W_δ. So it is worth taking a moment to develop some intuition for Condition 1. We have already seen in Corollary 1 that $\hat{\theta}_{\mathrm{IJ}}$ is, asymptotically, a good approximation for leave-k-out CV uniformly in W. We now discuss some additional cases: first, a worst-case example for which $\hat{\theta}_{\mathrm{IJ}}$ is not expected to work, second the bootstrap, and finally we revisit leave-one-out cross validation in the context of these other two methods.

First, consider a pathological example. Let W_{full} be the set of all weight vectors that sum to N. Let $n^* = \max_{n\in[N]} \left\|g_n\left(\hat{\theta}_1\right)\right\|_1$ be the index of the gradient term with the largest L_1 norm, and let $w_{n^*} = N$ and $w_n = 0$ for $n \neq n^*$. Then

$$\begin{aligned}&\sup_{\theta\in\Omega_\theta}\left\|\frac{1}{N}\sum_{n=1}^{N}(w_n - 1)\,g_n(\theta)\right\|_1 \\ &\quad= \sup_{\theta\in\Omega_\theta}\left\|g_{n^*}(\theta) - \frac{1}{N}\sum_{n=1}^{N} g_n(\theta)\right\|_1 \geq \left\|g_{n^*}\left(\hat{\theta}_1\right)\right\|_1.\end{aligned}$$

(The last inequality uses the fact that $G\left(\hat{\theta}_1, 1_w\right) = 0$.) In this case, unless the largest gradient, $\left\|g_{n^*}\left(\hat{\theta}_1\right)\right\|_1$, is small, Condition 1 will not be satisfied for small δ, and we would not expect $\hat{\theta}_{\mathrm{IJ}}$ to be a good estimate for $\hat{\theta}(w)$ for all $w \in W_{full}$. The class W_{full} is too expressive. In the language of Condition 1, for some small fixed δ, W_δ will be some very restricted subset of W_{full} in most realistic situations.

Now, suppose that we are using B bootstrap weights, $w_b^* \overset{iid}{\sim} \text{Multinomial}\left(N, N^{-1}\right)$ for $b = 1, ..., B$, and analyzing an optimization problem as defined in Section 2.1. For a given w_b^*, a dataset $x_1^*, ..., x_N^*$ formed by taking $w_{b,n}^*$ copies of datapoint x_n is equivalent in distribution to N IID samples with replacement from the

empirical distribution on $(x_1, ..., x_N)$. In this notation, we then have

$$\frac{1}{N}\sum_{n=1}^{N}\left(w_b^*-1\right)g_n\left(\theta\right)= \\ \frac{1}{N}\sum_{n=1}^{N}\frac{\partial f\left(\theta,x_n^*\right)}{\partial\theta}-\frac{1}{N}\sum_{n=1}^{N}\frac{\partial f\left(\theta,x_n\right)}{\partial\theta}.$$

In this case, Condition 1 is a uniform bound on a centered empirical process of derivatives of the objective function. Note that estimating sample variances by applying the IJ with bootstrap weights is equivalent to the ordinary delta method based on an asymptotic normal approximation [Efron, 1982, Chapter 21]. In order to provide an approximation to the bootstrap that retains benefits (such as the faster-than-normal convergence to the true sampling distribution described by Hall [2013]), one must consider higher-ordered Taylor expansions of $\hat{\theta}(w)$. We leave this for future work.

Finally, let us return to leave-one-out CV. In this case, $w_n - 1$ is nonzero for exactly one entry. Again, we can choose to leave out the adversarially-chosen n^* as in the first pathological example. However, unlike the pathological example, the leave-one-out CV weights are constrained to be closer to 1_w—specifically, we set $w_{n^*} = 0$, and let w be one elsewhere. Then Condition 1 requires $\sup_{\theta\in\Omega_\theta}\left\|\frac{1}{N}g_{n^*}\left(\theta\right)\right\|_1 \leq \delta$. In contrast to the pathological example, this supremum will get smaller as N increases as long as $\left\|g_{n^*}\left(\theta\right)\right\|_1$ grows more slowly than N. For this reason, we expect leave-one-out (and, indeed, leave-k-out for fixed k) to be accurately approximated by $\hat{\theta}_{\mathrm{IJ}}$ in many cases of interest, as stated in Corollary 1.

4 Related Work

Although the idea of forming a linear approximation to the re-weighting of an M-estimator has a long history, we nevertheless contribute in a number of ways. By limiting ourselves to approximating the exact reweighting on a particular dataset, we both loosen the strict requirements from the statistical literature and generalize the existing results from the machine learning literature.

The jackknife is often favored over the IJ in the statistics literature because of the former's simple computational approach, as well as perceived difficulties in calculating the necessary derivatives when some of the parameters are implicitly defined via optimization [Shao and Tu, 2012, Chapter 2.1] (though exceptions exist; see, e.g., Wager et al. [2014]). The brute-force approach of the jackknife is, however, a liability in large-scale machine learning problems, which are generally extremely expensive to re-optimize. Furthermore, and critically, the complexity and tedium of calculating the necessary derivatives is entirely eliminated by modern automatic differentiation [Baydin et al., 2017, Maclaurin et al., 2015].

Our work is based on the proof of the Fréchet differentiability of M-estimators of Clarke [1983]. In classical statistics, Fréchet differentiability is typically used

to describe the asymptotic behavior of functionals of the empirical distribution in terms of a functional [Mises, 1947, Fernholz, 1983]. Since Clarke [1983] was motivated by such asymptotic questions, he studied the Fréchet derivative evaluated at a continuous probability distribution for function classes that included delta functions. This focus led to the requirement of a bounded gradient. However, unbounded gradients are ubiquitous in both statistics and machine learning, and an essential contribution of the current paper is to remove the need for bounded gradients.

There exist proofs of the consistency of the (non-infinitesimal) jackknife that allow for unbounded gradients. For example, it is possible that the proofs of Reeds [1978], which require a smoothness assumption similar to our Assumption 4, could be adapted to the IJ. However, the results of Reeds [1978]—as well as those of Clarke [1983] and subsequent applications such as those of Shao and Tu [2012]—are asymptotic and applicable only to IID data. By providing finite sample results for a fixed dataset and weight set, we are able to provide a template for proving accuracy bounds for more generic probability distributions and re-weighting schemes.

A number of recent machine learning papers have derived approximate linear versions of leave-one-out estimators. Koh and Liang [2017] consider approximating the effect of leaving out one observation at a time to discover influential observations and construct adversarial examples, but provide little supporting theory. Beirami et al. [2017] provide rigorous proofs for an approximate leave-one-out CV estimator; however, their estimator requires computing a new inverse Hessian for each new weight at the cost of a considerable increase in computational complexity. Like the classical statistics literature, Beirami et al. [2017] assume that the gradients are bounded for all N. When $\|g\|_\infty^2$ in Corollary 1 is finite for all N, we achieve the same N^{-2} rate claimed by Beirami et al. [2017] for leave-one-out CV although we use only a single matrix inverse. Rad and Maleki [2018] also approximate leave-one-out CV, and prove tighter bounds for the error of their approximation than we do, but their work is customized to leave-one-out CV and makes much more restrictive assumptions (e.g., Gaussianity).

5 Simulated Experiments

We begin the empirical demonstration of our method on two simple generalized linear models: logistic and Poisson regression.[1] In each case, we generate a synthetic dataset $Z = \{(x_n, y_n)\}_{n=1}^N$ from parameters (θ, b), where $\theta \in \mathbb{R}^{100}$ is a vector of regression coefficients and $b \in \mathbb{R}$ is a bias term. In each experiment, $x_n \in \mathbb{R}^{100}$ is drawn from a multivariate Gaussian, and y_n is a scalar drawn from a Bernoulli distribution with the logit link or from a Poisson distribution with the exponential link.

[1] Leave-one-out CV may not be the most appropriate estimator of generalization error in this setting [Rosset and Tibshirani, 2018], but this section is intended only to provide simple illustrative examples.

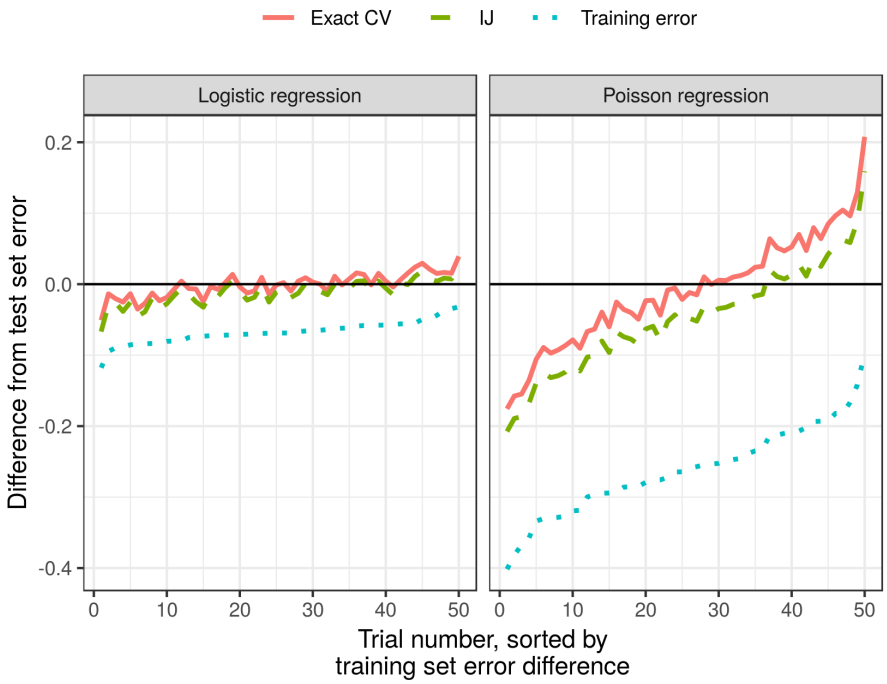


Figure 1: Simulated data: accuracy results.

For a ground truth, we generate a large test set with $N = 100{,}000$ datapoints to measure the true generalization error. We show in Fig. 1 that, over 50 randomly generated datasets, our approximation consistently underestimates the actual error predicted by exact leave-one-out CV; however, the difference is small relative to the improvements they both make over the error evaluated on the training set.

Fig. 2 shows the relative timings of our approximation and exact leave-one-out CV on logistic regression with datasets of increasing size. The time to run our approximation is roughly an order of magnitude smaller.

6 Genomics Experiments

We now consider a genomics application in which we use CV to choose the degree of a spline smoother when clustering time series of gene expression data. Code and instructions to reproduce our results can be found in the git repository rgiordan/AISTATS2019SwissArmyIJ. The application is also described in detail in Appendix B.

We use a publicly available data set of mice gene expression [Shoemaker et al., 2015] in which mice were infected with influenza virus, and gene expression was

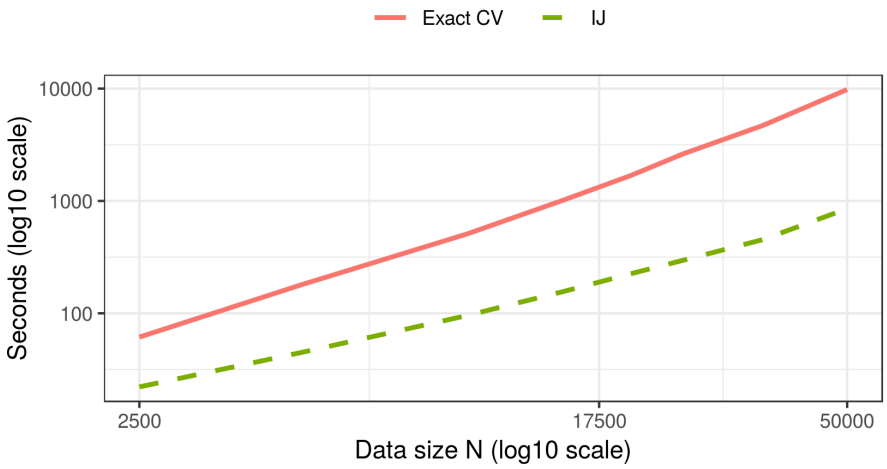


Figure 2: Simulated data: timing results.

assessed several times after infection. The observed data consists of expression levels y_{gt} for genes $g = 1, \ldots, n_g$ and time points $t = 1, \ldots, n_t$. In our case $n_g = 1000$ and $n_t = 14$. Many genes behave the same way; thus, clustering the genes by the pattern of their behavior over time allows dimensionality reduction that can facilitate interpretation. Consequently, we wish to first fit a smoothed regression line to each gene and then cluster the results. Following Luan and Li [2003], we model the time series as a gene-specific constant additive offset plus a B-spline basis of degree 3, and the task is to choose the B-spline basis degrees of freedom using cross-validation on the time points.

Our analysis runs in two stages—first, we regress the genes on the spline basis, and then we cluster a transformed version of the regression fits. By modeling in two stages, we both speed up the clustering and allow for the use of flexible transforms of the fits. We are interested in choosing the smoothing parameter using CV on the time points. Both the time points and the smoothing parameter enter the regression objective directly, but they affect the clustering objective only through the optimal regression parameters. Because the optimization proceeds in two stages, the fit is not the optimum of any single objective function. However, it can still be represented as an M-estimator (see Appendix B).

We implemented the model in `scipy` [Jones et al., 2001] and computed all derivatives with `autograd` [Maclaurin et al., 2015]. We note that the match between "exact" cross-validation (removing time points and re-optimizing) and the IJ was considerably improved by using a high-quality second-order optimization method. In particular, for these experiments, we employed the Newton conjugate-gradient trust region method [Wright and Nocedal, 1999, Chapter 7.1] as implemented by the method `trust-ncg` in `scipy.optimize`, preconditioned by the Cholesky decomposition of an inverse Hessian calculated at an initial

approximate optimum. The Hessian used for the preconditioner was with respect to the clustering parameters only and so could be calculated quickly, in contrast to the H_1 matrix used for the IJ, which includes the regression parameters as well. We found that first-order or quasi-Newton methods (such as BFGS) often got stuck or terminated at points with fairly large gradients. At such points our method does not apply in theory nor, we found, very well in practice.

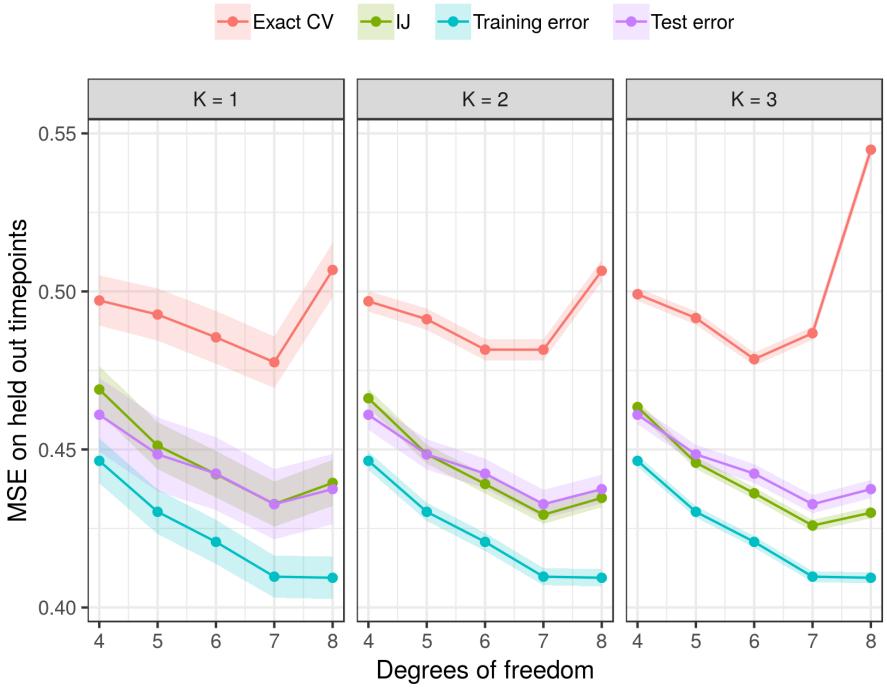


Figure 3: Genomics data: accuracy results.

Fig. 3 shows that the IJ is a reasonably good approximation to the test set error.[2] In particular, both the IJ and exact CV capture the increase in test error for $df = 8$, which is not present in the training error. Thus we see that, like exact CV, the IJ is able to prevent overfitting. Though the IJ underestimates exact CV, we note that it differs from exact CV by no more than exact CV itself differs from the true quantity of iterest, the test error.

The timing results for the genomics experiment are shown in Fig. 4. For this particular problem with approximately $39,000$ parameters (the precise number depends on the degrees of freedom), finding the initial optimum takes about 42 seconds. The cost of finding the initial optimum is shared by exact CV and the

[2]In fact, in this case, the IJ is a better predictor of test set error than exact CV. However, the authors have no reason at present to believe that the IJ is a better predictor of test error than exact CV in general.

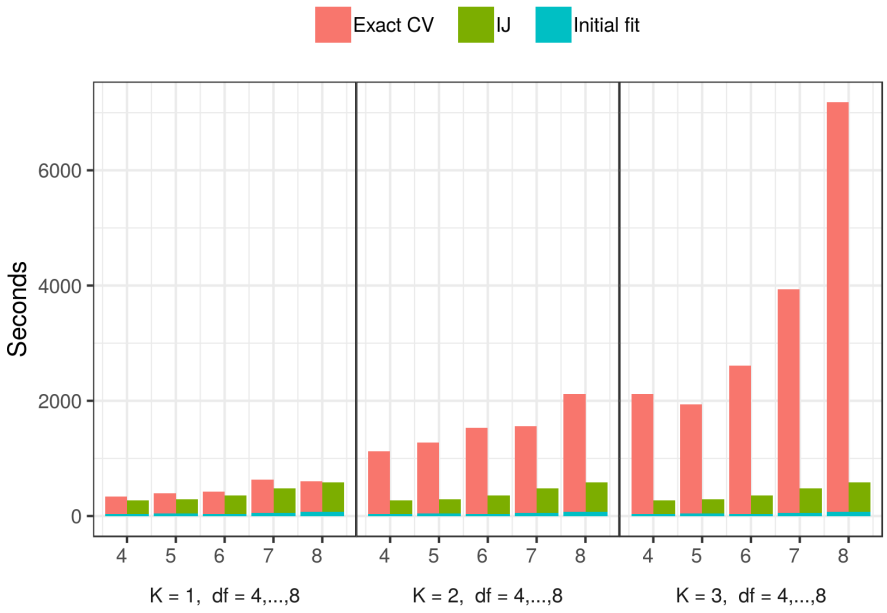


Figure 4: Genomics data: timing results.

IJ, and, as shown in Fig. 4, is a small proportion of both.

The principle time cost of the IJ is the computation of H_1. Computing and inverting a dense matrix of size $39,000$ would be computationally prohibitive. But, for the regression objective, H_1 is extremely sparse and block diagonal, so computing H_1 in this case took only around 360 seconds. Inverting H_1 took negligible time. Once we have H_1^{-1}, obtaining the subsequent IJ approximations is nearly instantaneous.

The cost of refitting the model for exact CV varies by degrees of freedom (increasing degrees of freedom increases the number of parameters) and the number of left-out points (an increasing number of left-out datapoints increases the number of refits). As can be seen in Fig. 4, for low degrees of freedom and few left-out points, the cost of re-optimizing is approximately the same as the cost of computing H_1. However, as the degrees of freedom and number of left-out points grow, the cost of exact CV increases to as much as an order of magnitude more than that of the IJ.

7 Conclusion

We recommend consideration of the Swiss Army infinitesimal jackknife for modern machine learning problems. The large size of modern data both increases the need for fast approximations and renders such approximations more accurate.

Furthermore, modern automatic differentiation renders many past practical difficulties obsolete. By stepping back from the strict requirements of classical statistical theory, we can see that the value of the infinitesimal jackknife extends beyond its traditional application areas, while retaining desirable generality in other respects.

Acknowledgements. We thank anonymous reviewers for their helpful comments and suggestions. We are grateful to Nelle Varoquaux for her help with the genomics experiments and to Pang Wei Koh for pointing out and helping to correct an error in an earlier version of our proofs. This research was supported in part by DARPA (FA8650-18-2-7832), an ARO YIP award, an NSF CAREER award, and the CSAIL-MSR Trustworthy AI Initiative. Ryan Giordano was supported by the Gordon and Betty Moore Foundation through Grant GBMF3834 and by the Alfred P. Sloan Foundation through Grant 2013-10-27 to the University of California, Berkeley. Runjing Liu was supported by the NSF Graduate Research Fellowship.

References

N. Agarwal, B. Bullins, and E. Hazan. Second-order stochastic optimization in linear time. *Journal of Machine Learning Research*, 2017.

A. Baydin, B. Pearlmutter, A. Radul, and J. Siskind. Automatic differentiation in machine learning: a survey. *Journal of Machine Learning Research*, 18(153): 1–153, 2017.

A. Beirami, M. Razaviyayn, S. Shahrampour, and V. Tarokh. On optimal generalizability in parametric learning. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3458–3468, 2017.

B. Clarke. Uniqueness and Fréchet differentiability of functional solutions to maximum likelihood type equations. *The Annals of Statistics*, 11(4):1196–1205, 1983.

R. Dudley. *Real analysis and probability*. Chapman and Hall/CRC, 2018.

B. Efron. *The Jackknife, the Bootstrap, and Other Resampling Plans*, volume 38. Society for Industrial and Applied Mathematics, 1982.

L. Fernholz. *Von Mises Calculus for Statistical Functionals*, volume 19. Springer Science & Business Media, 1983.

P. Hall. *The Bootstrap and Edgeworth Expansion*. Springer Science & Business Media, 2013.

L. Jaeckel. The infinitesimal jackknife, memorandum. Technical report, MM 72-1215-11, Bell Lab. Murray Hill, NJ, 1972.

E. Jones, T. Oliphant, P. Peterson, et al. SciPy: Open source scientific tools for Python, 2001. URL `http://www.scipy.org/`.

R. W. Keener. *Theoretical Statistics: Topics for a Core Course*. Springer, 2011.

P. W. Koh and P. Liang. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning (ICML)*, 2017.

Y. Luan and H. Li. Clustering of time-course gene expression data using a mixed-effects model with B-splines. *Bioinformatics*, 19(4):474–482, 2003.

D. Maclaurin, D. Duvenaud, and R. P. Adams. Autograd: Effortless gradients in numpy. In *International Conference on Machine Learning (ICML) AutoML Workshop*, 2015.

R. Mises. On the asymptotic distribution of differentiable statistical functions. *The Annals of Mathematical Statistics*, 18(3):309–348, 1947.

K. R. Rad and A. Maleki. A scalable estimate of the extra-sample prediction error via approximate leave-one-out. *arXiv Preprint*, January 2018.

J. A. Reeds. Jackknifing maximum likelihood estimates. *The Annals of Statistics*, pages 727–739, 1978.

S. Rosset and R. J. Tibshirani. From fixed-X to random-X regression: Bias-variance decompositions, covariance penalties, and prediction error estimation. *Journal of the American Statistical Association*, 2018.

J. Schott. *Matrix Analysis for Statistics*. John Wiley & Sons, 2016.

J. Shao. Differentiability of statistical functionals and consistency of the jackknife. *The Annals of Statistics*, pages 61–75, 1993.

J. Shao and D. Tu. *The Jackknife and Bootstrap*. Springer Series in Statistics, 2012.

J. E. Shoemaker, S. Fukuyama, A. J. Eisfeld, D. Zhao, E. Kawakami, S. Sakabe, T. Maemura, T. Gorai, H. Katsura, Y. Muramoto, S. Watanabe, T. Watanabe, K. Fuji, Y. Matsuoka, H. Kitano, and Y. Kawaoka. An ultrasensitive mechanism regulates influenza virus-induced inflammation. *PLoS Pathogens*, 11(6):1–25, 2015.

S. Wager, T. Hastie, and B. Efron. Confidence intervals for random forests: The jackknife and the infinitesimal jackknife. *The Journal of Machine Learning Research*, 15(1):1625–1651, 2014.

S. Wright and J. Nocedal. *Numerical Optimization*, volume 35. 1999.

A Detailed assumptions, lemmas, and proofs

A.1 Tools

We begin by stating two general propositions that will be useful. First, we show that a version of Cauchy-Schwartz can be applied to weighted sums of tensors.

Proposition 1. *Tensor array version of Hölder's inequality. Let w be an array of scalars and let $a = (a_1, ..., a_N)$ be an array of tensors, were each a_n is indexed by $i = 1, \ldots, D_A$ (i may be a multi-index—e.g., if A is a $D \times D$ matrix, then $i = (j, k)$, for $j, k \in [D]$ and $D_A = D^2$). Let $p, q \in [1, \infty]$ be two numbers such that $p^{-1} + q^{-1} = 1$. Then*

$$\left\| \frac{1}{N} \sum_{n=1}^{N} w_n a_n \right\|_1 \leq \frac{D_A^{\frac{1}{p}}}{N} \|w\|_p \|a\|_q .$$

In particular, with $p = q = 2$,

$$\left\| \frac{1}{N} \sum_{n=1}^{N} w_n a_n \right\|_1 \leq \sqrt{D_A} \frac{\|w\|_2}{\sqrt{N}} \frac{\|a\|_2}{\sqrt{N}}.$$

Proof. The conclusion follows from the ordinary Hölder's inequality applied term-wise to n and Jensen's inequality applied to the indices i.

$$\begin{aligned}
\left\| \frac{1}{N} \sum_{n=1}^{N} w_n a_n \right\|_1 &= \sum_{i=1}^{D_A} \left| \frac{1}{N} \sum_{n=1}^{N} w_n \left(a_n\right)_i \right| \\
&\leq \frac{1}{N} \sum_{i=1}^{D_A} \left| \left(\sum_{n=1}^{N} |w_n|^p \right)^{\frac{1}{p}} \left(\sum_{n=1}^{N} |(a_n)_i|^q \right)^{\frac{1}{q}} \right| \text{(Hölder)} \\
&= \frac{1}{N} \|w\|_p \frac{D_A}{D_A} \sum_{i=1}^{D_A} \left(\sum_{n=1}^{N} |(a_n)_i|^q \right)^{\frac{1}{q}} \\
&\leq \frac{1}{N} \|w\|_p D_A \left(\frac{1}{D_A} \sum_{i=1}^{D_A} \sum_{n=1}^{N} |(a_n)_i|^q \right)^{\frac{1}{q}} \text{ (Jensen applied to } i) \\
&= \frac{1}{N} \|w\|_p D_A \left(\frac{1}{D_A} \sum_{n=1}^{N} \|a_n\|_q^q \right)^{\frac{1}{q}} \\
&= \frac{1}{N} \|w\|_p D_A^{1-\frac{1}{q}} \|a\|_q \\
&= \frac{D_A^{\frac{1}{p}}}{N} \|w\|_p \|a\|_q .
\end{aligned}$$

□

Next, we prove a relationship between the term-wise difference between matrices and the difference between their operator norms. It is well-known that the minimum eigenvalue of a non-singular matrix is continuous in the entries of the matrix. In the next proposition, we quantify this continuity for the L_1 norm.

Proposition 2. *Let A and B be two square matrices of the same size. Let $\left\|A^{-1}\right\|_{op} \leq C_{op}$ for some finite C_{op}, and Then*

$$\|A - B\|_1 \leq \frac{1}{2}(C_{op})^{-1} \quad \Rightarrow \quad \left\|B^{-1}\right\|_{op} \leq 2C_{op}.$$

Proof. We will use the results stated in Theorem 4.29 of Schott [2016] and the associated discussion in Example 4.14, which establish the following result. Let A be a square, nonsigular matrix, and let I be the identity matrix of the same size. Let $\|\cdot\|$ denote any matrix norm satisfying $\|I\| = 1$. Let D be a matrix of the same size as A satisfying

$$\left\|A^{-1}\right\| \|D\| \leq 1. \tag{2}$$

Then

$$\left\|A^{-1} - (A + D)^{-1}\right\| \leq \frac{\left\|A^{-1}\right\| \|D\|}{1 - \|A^{-1}\|\|D\|\|} \left\|A^{-1}\right\|. \tag{3}$$

We will apply equation (3) using the operator norm $\|\cdot\|_{op}$, for which $\|I\|_{op} = 1$ and with $D := B - A$. Because $\left\|A^{-1}\right\|_{op} \leq C_{op}$, A is invertible.

Assume that $\|A - B\|_1 \leq \frac{1}{2}(C_{op})^{-1}$. First, note that

$$\begin{aligned}
\left\|A^{-1}\right\|_{op} \|D\|_{op} &= \left\|A^{-1}\right\|_{op} \|B - A\|_{op} \\
&\leq \left\|A^{-1}\right\|_{op} \|B - A\|_1 \qquad \text{(ordering of matrix norms)} \\
&\leq C_{op} \frac{1}{2}(C_{op})^{-1} \qquad \text{(by assumption)} \\
&= \frac{1}{2} < 1, \qquad (4)
\end{aligned}$$

so equation (2) is satisfied and we can apply equation (3). Then

$$\begin{aligned}
\left\|B^{-1}\right\|_{op} &\leq \left\|B^{-1} - A^{-1}\right\|_{op} + \left\|A^{-1}\right\|_{op} \qquad \text{(triangle inequality)} \\
&\leq \frac{\left\|A^{-1}\right\|_{op} \|B - A\|_{op}}{1 - \|A^{-1}\|_{op} \|B - A\|_{op}} \left\|A^{-1}\right\|_{op} + \left\|A^{-1}\right\|_{op} \qquad \text{(equation (3))} \\
&\leq \frac{\frac{1}{2}}{1 - \frac{1}{2}} \left\|A^{-1}\right\|_{op} + \left\|A^{-1}\right\|_{op} \qquad \text{(equation (4))} \\
&\leq 2C_{op}. \qquad \text{(by assumption)}
\end{aligned}$$

□

Next, we define the quantities needed to make use of the integral form of the Taylor series remainder.[3]

Proposition 3. *For any $\theta \in \Omega_\theta$ and any $\tilde{w} \in W$,*

$$G(\theta, \tilde{w}) - G(\hat{\theta}_1, \tilde{w}) = \left(\int_0^1 H(\hat{\theta}_1 + t(\theta - \hat{\theta}_1), w) dt \right) \left(\theta - \hat{\theta}_1 \right)$$

Proof. Let $G_d(\theta, \tilde{w})$ denote the d-th component of the vector $G(\theta, \tilde{w})$, and define the function $f_d(t) := G_d(\hat{\theta}_1 + t(\hat{\theta}_1 - \theta), \tilde{w})$. The proposition follows by taking the integral remainder form of the zero-th order Taylor series expansion of $f_d(t)$ around $t = 0$ [Dudley, 2018, Appendix B.2], and stacking the result into a vector. □

The Taylor series residual of Proposition 3 will show up repeatedly, so we will give it a concise name in the following definition.

Definition 4. For a fixed weight w and a fixed parameter θ, define the Hessian integral

$$\tilde{H}(\theta, w) := \int_0^1 H(\hat{\theta}_1 + t(\theta - \hat{\theta}_1), w) dt.$$

A.2 Lemmas

We now prove some useful consequences of our assumptions. The proof roughly proceeds for all $w \in W_\delta$ by the following steps:

1. When δ is small we can make $\left\| \hat{\theta}(w) - \hat{\theta}_1 \right\|_2$ small. (Lemma 3 below.)
2. When $\left\| \theta - \hat{\theta}_1 \right\|_2$ is small, then the derivatives $H(\theta, w)$ are close to their optimal value, $H\left(\hat{\theta}_1, 1_w\right)$. (Lemma 4 and Lemma 5 below.)
3. When the derivatives are close to their optimal values, then $H(\theta, w)$ is uniformly non-singular. (Lemma 6 below.)
4. When the derivatives are close to their optimal values and $H(\theta, w)$ is uniformly non-singular we can control the error in $\hat{\theta}_{\mathrm{IJ}} - \hat{\theta}(w)$ in terms of δ. (Theorem 2 below.)

We begin by showing that the difference between $\hat{\theta}(w)$ and $\hat{\theta}_1$ for $w \in W_\delta$ can be made small by making δ from Condition 1 small. First, however, we need to prove that operator norm bounds on $H(\theta, w)$ also apply to the integrated Hessian $\tilde{H}(\theta, w)$.

[3] We are indebted to Pang Wei Koh for pointing out the need to use the integral form of the remainder for Taylor series expansions of vector-valued functions.

Lemma 2. *Invertibility of the integrated Hessian. If, for some domain* Ω *and some constant* C, $\sup_{\theta\in\Omega}\left\|H(\theta,w)^{-1}\right\|_{op}\leq C$, *then* $\sup_{\theta\in\Omega}\left\|\tilde{H}(\theta,w)^{-1}\right\|_{op}\leq C$.

Proof. By definition of the operator norm,

$$\begin{aligned}\left\|\tilde{H}(\theta,w)^{-1}\right\|_{op}^{-1} &= \min_{v\in\mathbb{R}^D:\|v\|_2=1} v^T\tilde{H}(\theta,w)v \\ &= \min_{v\in\mathbb{R}^D:\|v\|_2=1}\int_0^1 v^T H(\hat{\theta}_1+t(\theta-\hat{\theta}_1),w)v dt \\ &\geq \int_0^1 \min_{v\in\mathbb{R}^D:\|v\|_2=1} v^T H(\hat{\theta}_1+t(\theta-\hat{\theta}_1),w)v dt \\ &\geq \inf_{\theta\in\Omega}\min_{v\in\mathbb{R}^D:\|v\|_2=1} v^T H(\theta,w)v \\ &\geq C^{-1}.\end{aligned}$$

The result follows by inverting both sides of the inequality. □

Lemma 3. *Small parameter changes. Under Assumptions 1—3 and Condition 1,*

$$\text{for all } w\in W_\delta,\quad \left\|\hat{\theta}(w)-\hat{\theta}_1\right\|_2\leq C_{op}\delta.$$

Proof. Applying Proposition 3 with $\theta=\hat{\theta}(w)$ and $\tilde{w}=1_w$ gives

$$G\left(\hat{\theta}(w),1_w\right)=G\left(\hat{\theta}_1,1_w\right)+\tilde{H}\left(\hat{\theta}(w),1_w\right)\left(\hat{\theta}(w)-\hat{\theta}_1\right).$$

By Assumption 2 and Lemma 2, $\sup_{\theta\in\Omega_\theta}\left\|\tilde{H}(\theta,1_w)^{-1}\right\|\leq C_{op}$. In particular, $\tilde{H}(\theta,1_w)$ is non-singular. A little manipulation, together with the fact that $G\left(\hat{\theta}(w),w\right)=G\left(\hat{\theta}_1,1_w\right)=0$ gives

$$\hat{\theta}(w)-\hat{\theta}_1=\tilde{H}\left(\hat{\theta}(w),1_w\right)^{-1}\left(G\left(\hat{\theta}(w),1_w\right)-G\left(\hat{\theta}(w),w\right)\right).$$

Taking the norm of both sides gives

$$\begin{aligned}\left\|\hat{\theta}(w)-\hat{\theta}_1\right\|_2 &= \left\|\tilde{H}\left(\hat{\theta}(w),1_w\right)^{-1}\left(G\left(\hat{\theta}(w),1_w\right)-G\left(\hat{\theta}(w),w\right)\right)\right\|_2 \\ &\leq \left\|\tilde{H}\left(\hat{\theta}(w),1_w\right)^{-1}\right\|_{op}\left\|\left(G\left(\hat{\theta}(w),1_w\right)-G\left(\hat{\theta}(w),w\right)\right)\right\|_2 \\ &\leq C_{op}\left\|G\left(\hat{\theta}(w),1_w\right)-G\left(\hat{\theta}(w),w\right)\right\|_2 \text{ (Lemma 2)} \\ &\leq C_{op}\left\|G\left(\hat{\theta}(w),1_w\right)-G\left(\hat{\theta}(w),w\right)\right\|_1 \text{ (relation between norms)} \\ &\leq C_{op}\sup_{\theta\in\Omega_\theta}\left\|G(\theta,1_w)-G(\theta,w)\right\|_1 \\ &\leq C_{op}\delta. \text{ (Condition 1).}\end{aligned}$$

□

Because we will refer to it repeatedly, we give the set of θ defined in Lemma 3 a name.

Definition 5. For a given δ, define the region around $\hat{\theta}_1$ given by Lemma 3 as

$$B_{C_{op}\delta} := \left\{\theta : \left\|\theta - \hat{\theta}_1\right\|_2 \leq C_{op}\delta\right\} \bigcap \Omega_\theta.$$

In other words, Lemma 3 states that Condition 1 implies $\hat{\theta}(w) \in B_{C_{op}\delta}$ when $w \in W_\delta$.

Next, we show that closeness in θ will mean closeness in $H(\theta, w)$.

Lemma 4. *Boundedness and continuity. Under Assumptions 1–5 and Condition 1,*

$$\text{for all } \theta \in B_{\Delta_\theta}, \quad \sup_{w\in W} \left\|H(\theta, w) - H\left(\hat{\theta}_1, w\right)\right\|_1 \leq DC_w L_h \left\|\theta - \hat{\theta}_1\right\|_2.$$

Proof. For $\theta \in B_{\Delta_\theta}$,

$$\begin{aligned}
\sup_{w\in W} \left\|H(\theta, w) - H\left(\hat{\theta}_1, w\right)\right\|_1 &= \sup_{w\in W} \left\|\frac{1}{N}\sum_{n=1}^{N} w_n \left(h_n(\theta) - h_n\left(\hat{\theta}_1\right)\right)\right\|_1 \text{ (by definition)} \\
&\leq D \sup_{w\in W} \frac{\|w\|_2}{\sqrt{N}} \frac{\left\|h(\theta) - h\left(\hat{\theta}_1\right)\right\|_2}{\sqrt{N}} \text{ (Proposition 1)} \\
&\leq DC_w \frac{\left\|h(\theta) - h\left(\hat{\theta}_1\right)\right\|_2}{\sqrt{N}} \text{ (Assumption 5)} \\
&\leq DC_w L_h \left\|\theta - \hat{\theta}_1\right\|_2 \text{ (Assumption 4 and } \theta \in B_{\Delta_\theta}).
\end{aligned}$$

□

We now combine Lemma 3 and Lemma 4 to show that $H(\theta, w)$ is close to its value at the solution $H\left(\hat{\theta}_1, 1_w\right)$ for sufficiently small δ and for all $\theta \in B_{C_{op}\delta}$.

Lemma 5. *Bounds for difference in parameters. Under Assumptions 1–5 and Condition 1, if $\delta \leq \Delta_\theta C_{op}^{-1}$, then*

$$\sup_{\theta\in B_{C_{op}\delta}} \sup_{w\in W_\delta} \left\|H(\theta, w) - H\left(\hat{\theta}_1, 1_w\right)\right\|_1 \leq \left(1 + DC_w L_h C_{op}\right)\delta.$$

Proof. By Lemma 3, $\delta \leq \Delta_\theta C_{op}^{-1}$ implies that $C_{op}\delta \leq \Delta_\theta$ and so $B_{C_{op}\delta} \subseteq B_{\Delta_\theta}$. Consequently, we can apply Lemma 4:

$$\begin{aligned}
\sup_{\theta\in B_{C_{op}\delta}} \sup_{w\in W_\delta} \left\|H(\theta, w) - H\left(\hat{\theta}_1, w\right)\right\|_1 &\leq \sup_{\theta\in B_{\Delta_\theta}} \sup_{w\in W_\delta} \left\|H(\theta, w) - H\left(\hat{\theta}_1, w\right)\right\|_1 \\
&\leq DC_w L_h \left\|\theta - \hat{\theta}_1\right\|_2 \text{ (Lemma 4)} \\
&\leq DC_w L_h C_{op}\delta \quad \text{(because } \theta \in B_{C_{op}\delta}).
\end{aligned}$$

Next, we can use this to write

$$\begin{aligned}
&\sup_{\theta\in B_{C_{op}\delta}}\sup_{w\in W_\delta}\left\|H\left(\theta,w\right)-H\left(\hat{\theta}_1,1_w\right)\right\|_1 \\
&\quad=\sup_{\theta\in B_{C_{op}\delta}}\sup_{w\in W_\delta}\left\|H\left(\theta,w\right)-H\left(\theta,1_w\right)+H\left(\theta,1_w\right)-H\left(\hat{\theta}_1,1_w\right)\right\|_1 \\
&\quad\leq\sup_{\theta\in B_{C_{op}\delta}}\sup_{w\in W_\delta}\left\|H\left(\theta,w\right)-H\left(\theta,1_w\right)\right\|_1+\sup_{\theta\in B_{C_{op}\delta}}\sup_{w\in W_\delta}\left\|H\left(\theta,1_w\right)-H\left(\hat{\theta}_1,1_w\right)\right\|_1 \\
&\quad\leq\sup_{\theta\in\Omega_\theta}\sup_{w\in W_\delta}\left\|H\left(\theta,w\right)-H\left(\theta,1_w\right)\right\|_1+\sup_{\theta\in B_{C_{op}\delta}}\sup_{w\in W_\delta}\left\|H\left(\theta,1_w\right)-H\left(\hat{\theta}_1,1_w\right)\right\|_1 \\
&\quad\leq\delta+\sup_{\theta\in B_{C_{op}\delta}}\sup_{w\in W_\delta}\left\|H\left(\theta,1_w\right)-H\left(\hat{\theta}_1,1_w\right)\right\|_1\ \text{(Condition 1)} \\
&\quad\leq\delta+DC_wL_hC_{op}\delta.
\end{aligned}$$

□

The constant that appears multiplying δ at the end of the proof of Lemma 5 will appear often in what follows, so we give it the special name C_{IJ} in Definition 3.

Note that Lemma 5 places a condition on how small δ must be in order for our regularity conditions to apply. Lemma 3 will guarantee that $\hat{\theta}\left(w\right)\in B_{C_{op}\delta}$, but if we are not able to make δ arbitrarily small in Condition 1, then we are not guaranteed to ensure that $B_{C_{op}\delta}\subseteq B_{\Delta_\theta}$, will not be able to assume Lipschitz continuity, and none of our results will apply.

Next, using Lemma 5, we can extend the operator bound on H_1^{-1} from Assumption 2 to $H\left(\theta,w\right)^{-1}$ for all $w\in W_\delta$.

Lemma 6. *Uniform invertibility of the Hessian. Under Assumptions 1–5 and Condition 1, if* $\delta\leq\min\left\{\Delta_\theta C_{op}^{-1},\frac{1}{2}C_{IJ}^{-1}C_{op}^{-1}\right\}$*, then*

$$\sup_{\theta\in B_{C_{op}\delta}}\sup_{w\in W_\delta}\left\|H\left(\theta,w\right)^{-1}\right\|_{op}\leq 2C_{op}.$$

Proof. By Assumption 2, $\left\|H\left(\hat{\theta}_1,1_w\right)^{-1}\right\|_{op}\leq C_{op}$. So by Proposition 2, it will suffice to select δ so that

$$\sup_{\theta\in B_{C_{op}\delta}}\sup_{w\in W_\delta}\left\|H\left(\theta,w\right)-H\left(\hat{\theta}_1,1_w\right)\right\|_1\leq\frac{1}{2}C_{op}^{-1}. \tag{5}$$

When we can apply Lemma 5, we have

$$\sup_{\theta\in B_{C_{op}\delta}}\sup_{w\in W_\delta}\left\|H\left(\theta,w\right)-H\left(\hat{\theta}_1,1_w\right)\right\|_1\leq C_{\text{IJ}}\delta.$$

So $H\left(\theta,w\right)$ will satisfy equation (5) if we can apply Lemma 5 and if

$$\delta\leq\frac{1}{2}C_{op}^{-1}C_{\text{IJ}}^{-1}.$$

To apply Lemma 5 we additionally require that $\delta \leq \Delta_\theta C_{op}^{-1}$. By taking $\delta \leq \min\left\{\Delta_\theta C_{op}^{-1}, \frac{1}{2} C_{op}^{-1} C_{\text{IJ}}^{-1}\right\}$, we satisfy equation (5) and the result follows. □

Next, we show that a version of Lemma 5 also applies to the integrated Hessian $\tilde{H}(\theta, w)$ when $\theta \in B_{C_{op}\delta}$.

Lemma 7. *Bounds for difference of the integrated Hessian. Under Assumptions 1–5 and Condition 1, if $\delta \leq \Delta_\theta C_{op}^{-1}$ and $\theta \in B_{C_{op}\delta}$,*

$$\sup_{w \in W_\delta} \left\| \tilde{H}(\theta, w) - H(\hat{\theta}_1, 1_w) \right\|_1 \leq \left(1 + DC_w L_h C_{op}\right) \delta.$$

Proof.

$$\begin{aligned}
&\sup_{w \in W_\delta} \left\| \tilde{H}(\theta, w) - H(\hat{\theta}_1, 1_w) \right\|_1 \\
&= \sup_{w \in W_\delta} \left\| \int_0^1 \left(H(\hat{\theta}_1 + t(\theta - \hat{\theta}_1), w) dt - H(\hat{\theta}_1, 1_w) \right) \right\|_1 && \text{(Definition 4)} \\
&\leq \sup_{w \in W_\delta} \int_0^1 \left\| H(\hat{\theta}_1 + t(\theta - \hat{\theta}_1), w) - H(\hat{\theta}_1, 1_w) \right\|_1 dt && \text{(Jensen's inequality)} \\
&\leq \sup_{\theta \in B_{C_{op}\delta}} \sup_{w \in W_\delta} \left\| H(\theta), w) - H(\hat{\theta}_1, 1_w) \right\|_1 \\
&\leq \left(1 + DC_w L_h C_{op}\right) \delta && \text{(Lemma 5)}
\end{aligned}$$

□

With these results in hand, the upper bound on δ will at last be sufficient to control the error terms in our approximation. For compactness, we give it the upper bound on δ the name Δ_δ in Definition 3.

Finally, we state a result that will allow us to define derivatives of $\hat{\theta}(w)$ with respect to w.

Lemma 8. *Inverse function theorem. Under Assumptions 1–5 and Condition 1, and for $\delta \leq \Delta_\delta$, there exists a continuous, differentiable function of w, $\hat{\theta}(w)$, such that, for all $w \in W$, $G\left(\hat{\theta}(w), w\right) = 0$.*

Proof. This follows from Lemma 6 and the implicit function theorem. □

By definition, $\hat{\theta}(1_w) = \hat{\theta}_1$.

A.3 Bounding the errors in a Taylor expansion

We are now in a position to use Assumptions 1–5 and Condition 1 to bound the error terms in a first-order Taylor expansion of $\hat{\theta}(w)$. We begin by simply calculating the derivative $d\hat{\theta}(w)/dw$.

Proposition 4. *For any $w \in W$ for which $H\left(\hat{\theta}(w), w\right)$ is invertible, and for any vector $a \in \mathbb{R}^N$,*

$$\frac{d\hat{\theta}(w)}{dw^T}|_w a = -H\left(\hat{\theta}(w), w\right)^{-1} G\left(\hat{\theta}(w), a\right).$$

Proof. Because $G\left(\hat{\theta}(w), w\right) = 0$ for all $w \in W$, by direct calculation,

$$\begin{aligned}
0 &= \frac{d}{dw^T} G\left(\hat{\theta}(w), w\right)|_w a \\
&= \left(\frac{\partial G}{\partial \theta^T}\frac{d\hat{\theta}}{dw^T} + \frac{\partial G}{\partial w^T}\right)|_w a \\
&= H\left(\hat{\theta}(w), w\right)\frac{d\hat{\theta}}{dw^T}|_w a + \left(\frac{\partial}{\partial w^T}\frac{1}{N}\sum_{n=1}^{N} w_n g_n(\theta)\right)|_w a \\
&= H\left(\hat{\theta}(w), w\right)\frac{d\hat{\theta}}{dw^T}|_w a + \frac{1}{N}\sum_{n=1}^{N} g_n\left(\hat{\theta}(w)\right) a \\
&= H\left(\hat{\theta}(w), w\right)\frac{d\hat{\theta}}{dw^T}|_w a + G\left(\hat{\theta}(w), a\right).
\end{aligned}$$

Because $H\left(\hat{\theta}(w), w\right)$ is invertible by assumption, the result follows. □

Definition 6. Define

$$\begin{aligned}
\hat{\theta}_{\mathrm{IJ}}(w) &:= \hat{\theta}_1 + \frac{d\hat{\theta}(w)}{dw^T}|_{1_w}(w - 1_w) \\
&= \hat{\theta}_1 - H_1^{-1} G\left(\hat{\theta}_1, w\right). \text{ (because } G\left(\hat{\theta}_1, 1_w\right) = 0)
\end{aligned}$$

$\hat{\theta}_{\mathrm{IJ}}(w)$ in Definition 6 is the first term in a Taylor series expansion of $\hat{\theta}(w)$ as a function of w. We want to bound the error, $\hat{\theta}_{\mathrm{IJ}}(w) - \hat{\theta}(w)$.

Theorem 2. *Under Assumptions 1–5 and Condition 1, when $\delta \leq \Delta_\delta$,*

$$\sup_{w \in W_\delta} \left\| \hat{\theta}_{IJ}(w) - \hat{\theta}(w) \right\|_2 \leq 2C_{op}^2 C_{IJ} \delta^2.$$

Proof. Applying Proposition 3 with $\theta = \hat{\theta}(w)$ and $\tilde{w} = w$, we have

$$0 = G\left(\hat{\theta}(w), w\right) = G\left(\hat{\theta}_1, w\right) + \tilde{H}\left(\hat{\theta}(w), w\right)\left(\hat{\theta}(w) - \hat{\theta}_1\right).$$

Because $\delta \in W_\delta$, Lemma 3 implies that $\hat{\theta}(w) \in B_{C_{op}\delta}$ so, Lemma 6 and Lemma

2 imply that $\tilde{H}\left(\hat{\theta}(w), w\right)$ is invertible and we can solve for $\hat{\theta}(w) - \hat{\theta}_1$.

$$\begin{aligned}
\hat{\theta}(w) - \hat{\theta}_1 &= -\tilde{H}\left(\hat{\theta}(w), w\right)^{-1} G\left(\hat{\theta}_1, w\right) \\
&= \left(-\tilde{H}\left(\tilde{\theta}, w\right)^{-1} + H\left(\hat{\theta}_1, 1_w\right)^{-1} - H\left(\hat{\theta}_1, 1_w\right)^{-1}\right) G\left(\hat{\theta}_1, w\right) \\
&= \left(H\left(\hat{\theta}_1, 1_w\right)^{-1} - \tilde{H}\left(\hat{\theta}(w), w\right)^{-1}\right) G\left(\hat{\theta}_1, w\right) + \hat{\theta}_{\mathrm{IJ}}(w) - \hat{\theta}_1.
\end{aligned}$$

Eliminating $\hat{\theta}_1$ and taking the supremum of both sides gives

$$\begin{aligned}
&\sup_{w \in W_\delta} \left\| \hat{\theta}_{\mathrm{IJ}}(w) - \hat{\theta}(w) \right\|_2 \\
&\quad = \sup_{w \in W_\delta} \left\| \left(H\left(\hat{\theta}_1, 1_w\right)^{-1} - \tilde{H}(\theta, w)^{-1}\right) G\left(\hat{\theta}_1, w\right) \right\|_2 \\
&\quad = \sup_{w \in W_\delta} \left\| \tilde{H}\left(\hat{\theta}(w), w\right)^{-1} \left(\tilde{H}\left(\hat{\theta}(w), w\right) - H\left(\hat{\theta}_1, 1_w\right)\right) H\left(\hat{\theta}_1, 1_w\right)^{-1} G\left(\hat{\theta}_1, w\right) \right\|_2 \\
&\quad \leq 2C_{op} \sup_{w \in W_\delta} \left\| \left(\tilde{H}\left(\hat{\theta}(w), w\right) - H\left(\hat{\theta}_1, 1_w\right)\right) H\left(\hat{\theta}_1, 1_w\right)^{-1} G\left(\hat{\theta}_1, w\right) \right\|_2 \text{ (Lemma 2)} \\
&\quad \leq 2C_{op} \sup_{w \in W_\delta} \left\| \tilde{H}\left(\hat{\theta}(w), w\right) - H\left(\hat{\theta}_1, 1_w\right) \right\|_{op} \left\| H\left(\hat{\theta}_1, 1_w\right)^{-1} G\left(\hat{\theta}_1, w\right) \right\|_2 \\
&\quad \leq 2C_{op} \sup_{w \in W_\delta} \left\| \tilde{H}\left(\hat{\theta}(w), w\right) - H\left(\hat{\theta}_1, 1_w\right) \right\|_1 \left\| H\left(\hat{\theta}_1, 1_w\right)^{-1} G\left(\hat{\theta}_1, w\right) \right\|_2 \text{ (ordering of matrix norms)} \\
&\quad \leq 2C_{op} C_{\mathrm{IJ}} \delta \sup_{w \in W_\delta} \left\| H\left(\hat{\theta}_1, 1_w\right)^{-1} G\left(\hat{\theta}_1, w\right) \right\|_2 \text{ (Lemma 7)} \\
&\quad \leq 2C_{op}^2 C_{\mathrm{IJ}} \delta \sup_{w \in W_\delta} \left\| G\left(\hat{\theta}_1, w\right) \right\|_2 \text{ (Assumption 2)} \\
&\quad = 2C_{op}^2 C_{\mathrm{IJ}} \delta \sup_{w \in W_\delta} \left\| G\left(\hat{\theta}_1, w\right) - G\left(\hat{\theta}_1, 1_w\right) \right\|_2 \text{ (because } G\left(\hat{\theta}_1, 1_w\right) = 0) \\
&\quad \leq 2C_{op}^2 C_{\mathrm{IJ}} \delta^2 \text{ (Condition 1).}
\end{aligned}$$

□

A.4 Use cases

First, let us state a simple condition under which Assumptions 1–4 hold. It will help to have a lemma for the Lipschitz continuity.

Lemma 9. *Derivative Cauchy Schwartz. Let $a(\theta) = (a_1(\theta), ..., a_N(\theta))$ be an array of tensors with multi-index $i \in [D_A]$, and let $\frac{\partial a(\theta)}{\partial \theta} = \left(\frac{\partial}{\partial \theta} a_1(\theta), ..., \frac{\partial}{\partial \theta} a_N(\theta)\right)$ be an array of tensors of size $D \times D_A$. Then*

$$\left\| \frac{\partial}{\partial \theta} \|a(\theta)\|_2 \right\|_2 \leq D_A \left\| \frac{\partial a}{\partial \theta} \right\|_2.$$

Proof. By direct calculation,

$$\begin{aligned}
\left\| \frac{\partial}{\partial \theta} \left\| a\left(\theta\right) \right\|_2^2 \right\|_2^2 &= \sum_{r=1}^{D} \left(\frac{\partial}{\partial \theta_r} \sum_{n=1}^{N} \sum_{i=1}^{D_A} a_{n,i}\left(\theta\right)^2 \right)^2 \\
&= \sum_{r=1}^{D} \left(\sum_{n=1}^{N} \sum_{i=1}^{D_A} 2 a_{n,i}\left(\theta\right) \frac{\partial a_{n,i}\left(\theta\right)}{\partial \theta_r} \right)^2 \\
&\leq \sum_{r=1}^{D} \left(2 \sum_{i=1}^{D_A} \left(\sum_{n=1}^{N} a_{n,i}\left(\theta\right)^2 \right)^{\frac{1}{2}} \left(\sum_{n=1}^{N} \left(\frac{\partial a_{n,i}\left(\theta\right)}{\partial \theta_r} \right)^2 \right)^{\frac{1}{2}} \right)^2 \\
&\leq \sum_{r=1}^{D} \left(2 D_A^2 \left(\frac{1}{D_A} \sum_{i=1}^{D_A} \sum_{n=1}^{N} a_{n,i}\left(\theta\right)^2 \right)^{\frac{1}{2}} \left(\frac{1}{D_A} \sum_{n=1}^{N} \left(\frac{\partial a_{n,i}\left(\theta\right)}{\partial \theta_r} \right)^2 \right)^{\frac{1}{2}} \right)^2 \\
&= 4 D_A^2 \left\| a \right\|_2^2 \sum_{r=1}^{D} \left\| \frac{\partial a}{\partial \theta_r} \right\|_2^2 \\
&= 4 D_A^2 \left\| a \right\|_2^2 \left\| \frac{\partial a}{\partial \theta} \right\|_2^2 .
\end{aligned}$$

By the chain rule,

$$\left\| \frac{\partial}{\partial \theta} \left\| a\left(\theta\right) \right\|_2 \right\|_2^2 = \frac{1}{4 \left\| a\left(\theta\right) \right\|_2^2} \left\| \frac{\partial}{\partial \theta} \left\| a\left(\theta\right) \right\|_2^2 \right\|_2^2 \leq D_A^2 \left\| \frac{\partial a}{\partial \theta} \right\|_2^2 .$$

□

Lemma 10. *Let $a\left(\theta\right) \in \mathbb{R}^{D \times D}$ be a continuously differentiable random matrix with a $D \times D \times D$ derivative tensor. (Note that the function, not θ, is random. For example, $\mathbb{E}\left[a\left(\theta\right)\right]$ is still a function of θ.) Suppose that $\mathbb{E}\left[\left\| a\left(\theta\right) \right\|_2\right]$ is finite for all $\theta \in \Omega_\theta$. Then, for all $\theta_1, \theta_2 \in \Omega_\theta$,*

$$\left| \mathbb{E}\left[\left\| a\left(\theta_1\right) \right\|_2\right] - \mathbb{E}\left[\left\| a\left(\theta_2\right) \right\|_2\right] \right| \leq \sqrt{\mathbb{E}\left[\sup_{\theta \in \Omega_\theta} \left\| \frac{\partial a\left(\theta\right)}{\partial \theta} \right\|_2^2 \right]} \left\| \theta_1 - \theta_2 \right\|_2 .$$

Proof. For any tensor a with multi-index i,

$$\begin{aligned}
\left\| \frac{\partial}{\partial \theta} \|a\|_2^2 \right\|_2^2 &= \sum_{r=1}^{D} \left(\frac{\partial}{\partial \theta_r} \|a\|_2^2 \right)^2 \\
&= \sum_{r=1}^{D} \left(\frac{\partial}{\partial \theta_r} \sum_{i=1}^{D_A} a_i^2 \right)^2 \\
&= \sum_{r=1}^{D} \left(2 \sum_{i=1}^{D_A} a_i \frac{\partial a_i}{\partial \theta_r} \right)^2 \\
&\leq 4 \sum_{r=1}^{D} \sum_{i=1}^{D_A} a_i^2 \sum_{i=1}^{D_A} \left(\frac{\partial a_i}{\partial \theta_r} \right)^2 \quad \text{(Cauchy-Schwartz)} \\
&= 4 \sum_{i=1}^{D_A} a_i^2 \sum_{r=1}^{D} \sum_{i=1}^{D_A} \left(\frac{\partial a_i}{\partial \theta_r} \right)^2 \\
&= 4 \|a\|_2^2 \left\| \frac{\partial a}{\partial \theta} \right\|_2^2 .
\end{aligned}$$

Consequently,

$$\begin{aligned}
\left\| \frac{\partial}{\partial \theta} \|a(\theta)\|_2 \right\|_2^2 &= \left\| \frac{1}{2 \|a(\theta)\|_2} \frac{\partial}{\partial \theta} \|a(\theta)\|_2^2 \right\|_2^2 \\
&= \frac{1}{4 \|a(\theta)\|_2^2} \left\| \frac{\partial}{\partial \theta} \|a(\theta)\|_2^2 \right\|_2^2 \\
&\leq \frac{4 \|a(\theta)\|_2^2}{4 \|a(\theta)\|_2^2} \left\| \frac{\partial}{\partial \theta} a(\theta) \right\|_2^2 \\
&= \left\| \frac{\partial a(\theta)}{\partial \theta} \right\|_2^2 .
\end{aligned}$$

So for any $\theta_1, \theta_2 \in \Omega_\theta$,

$$\begin{aligned}
|\mathbb{E}[\|a(\theta_1)\|_2] - \mathbb{E}[\|a(\theta_2)\|_2]| &\leq \mathbb{E}[|\|a(\theta_1)\|_2 - \|a(\theta_2)\|_2|] \\
&\leq \mathbb{E}\left[\left(\sup_{\theta \in \Omega_\theta} \left\| \frac{\partial}{\partial \theta} \|a(\theta)\|_2 \right\|_2 \right) \right] \|\theta_1 - \theta_2\|_2 \quad (\theta \text{ is not random}) \\
&\leq \mathbb{E}\left[\left(\sup_{\theta \in \Omega_\theta} \left\| \frac{\partial a(\theta)}{\partial \theta} \right\|_2 \right) \right] \|\theta_1 - \theta_2\|_2 \\
&\leq \sqrt{\mathbb{E}\left[\sup_{\theta \in \Omega_\theta} \left\| \frac{\partial a(\theta)}{\partial \theta} \right\|_2^2 \right]} \|\theta_1 - \theta_2\|_2 .
\end{aligned}$$

The result follows. Note that the bound still holds (though vacuously) if $\mathbb{E}\left[\sup_{\theta \in \Omega_\theta} \left\| \frac{\partial a(\theta)}{\partial \theta} \right\|_2^2 \right]$ is infinite. □

Proposition 5. *Let Ω_θ be a compact set. Let $g_n(\theta)$ be twice continuously differentiable IID random functions. Define*

$$h_n(\theta) := \frac{\partial g_n(\theta)}{\partial \theta}$$
$$r_n(\theta) := \frac{\partial^2 g_n(\theta)}{\partial \theta \partial \theta},$$

where $r_n(\theta)$ is a $D \times D \times D$ tensor. Assume that

1a) $\mathbb{E}\left[\sup_{\theta\in\Omega_\theta} \|g_n(\theta)\|_2^2\right] < \infty$*;*

1b) $\mathbb{E}\left[\sup_{\theta\in\Omega_\theta} \|h_n(\theta)\|_2^2\right] < \infty$*;*

1c) $\mathbb{E}\left[\sup_{\theta\in\Omega_\theta} \|r_n(\theta)\|_2^2\right] < \infty$;

2) $\mathbb{E}[h_n(\theta)]$ *is non-singular for all* $\theta \in \Omega_\theta$*;*

3) We can exchange expectation and differentiation.

Then $\lim_{N\to\infty} P(\text{Assumptions 1–4 hold}) = 1$.

Proof. The proof follows from Theorems 9.1 and 9.2 of Keener [2011]. We will first show that the expected values of the needed functions satisfy Assumptions 1–4 , and then that the sample versions converge uniformly.

By Jensen's inequality,

$$\mathbb{E}\left[\sup_{\theta\in\Omega_\theta} \|g_n(\theta)\|_2\right] = \mathbb{E}\left[\sqrt{\sup_{\theta\in\Omega_\theta} \|g_n(\theta)\|_2^2}\right] \leq \sqrt{\mathbb{E}\left[\sup_{\theta\in\Omega_\theta} \|g_n(\theta)\|_2^2\right]}.$$

Also, for the i^{th} component of $g_n(\theta)$

$$\mathbb{E}\left[\sup_{\theta\in\Omega_\theta} |g_{n,i}(\theta)|\right] \leq \mathbb{E}\left[\sup_{\theta\in\Omega_\theta} \|g_n(\theta)\|_\infty\right] \leq \mathbb{E}\left[\sup_{\theta\in\Omega_\theta} \|g_n(\theta)\|_2\right].$$

By Theorem 9.1 of Keener [2011], $\mathbb{E}\left[\|g_n(\theta)\|_2^2\right]$, $\mathbb{E}[\|g_n(\theta)\|_2]$, and $\mathbb{E}[g_n(\theta)]$ are continuous functions of θ, and because Ω_θ is compact, they are each bounded. Similar reasoning applies to $h_n(\theta)$ and $r_n(\theta)$. Consequently we can define

$$\sup_{\theta\in\Omega_\theta} \mathbb{E}\left[\|g_n(\theta)\|_2^2\right] =: Q_g^2 < \infty$$
$$\sup_{\theta\in\Omega_\theta} \mathbb{E}\left[\|h_n(\theta)\|_2^2\right] =: Q_h^2 < \infty.$$

Below, these constants will be used to satisfy Assumption 1 and Assumption 3 with high probability.

Because Ω_θ is compact, $\mathbb{E}[h_n(\theta)]$ is continuous, $\mathbb{E}[h_n(\theta)]$ is non-singular, and the operator norm is a continuous function of $\mathbb{E}[h_n(\theta)]$, we can also define

$$\sup_{\theta\in\Omega_\theta} \left\|\mathbb{E}[h_n(\theta)]^{-1}\right\|_{op} =: Q_{op} < \infty.$$

Below, this constant be used to satisfy Assumption 2 with high probability.

Finally, we turn to the Lipschitz condition. Lemma 10 implies that

$$\left|\mathbb{E}\left[\left\|h_n\left(\theta_1\right)\right\|_2\right] - \mathbb{E}\left[\left\|h_n\left(\theta_2\right)\right\|_2\right]\right| \leq \sqrt{\mathbb{E}\left[\sup_{\theta\in\Omega_\theta}\left\|r_n\left(\theta\right)\right\|_2^2\right]}\left\|\theta_1 - \theta_2\right\|_2.$$

Define

$$\Lambda_h = \sqrt{\mathbb{E}\left[\sup_{\theta\in\Omega_\theta}\left\|r_n\left(\theta\right)\right\|_2^2\right]},$$

so that we have shown that $\mathbb{E}\left[\left\|h_n\left(\theta\right)\right\|_2\right]$ is Lipschitz in Ω_θ with constant Λ_h, which is finite by assumption.

We have now shown, essentially, that the expected versions of the quantities we wish to control satisfy Assumptions 1–4 with $N = 1$. We now need to show that the sample versions satisfy Assumptions 1–4 with high probability, which will follow from the fact that the sample versions converge uniformly to their expectations by Theorem 9.2 of Keener [2011].

First, observe that Assumption 1 holds with probability one by assumption. For the remaining assumption choose an $\epsilon > 0$, and define

$$\begin{aligned} C_g &:= \sqrt{Q_g^2 + \epsilon} \\ C_h &:= \sqrt{Q_h^2 + \epsilon} \\ C_{op} &:= 2Q_{op} \\ L_h &:= \sqrt{D^4\Lambda_h^2 + \epsilon}. \end{aligned}$$

By Keener [2011] Theorem 9.2,

$$\sup_{\theta\in\Omega_\theta}\left|\frac{1}{N}\sum_{n=1}^{N}\left\|g_n\left(\theta\right)\right\|_2^2 - \mathbb{E}\left[\left\|g_n\left(\theta\right)\right\|_2^2\right]\right| \xrightarrow[N\to\infty]{p} 0.$$

Because

$$\sup_{\theta\in\Omega_\theta}\left|\frac{1}{N}\sum_{n=1}^{N}\left\|g_n\left(\theta\right)\right\|_2^2\right| > Q_g^2 + \epsilon \geq \sup_{\theta\in\Omega_\theta}\mathbb{E}\left[\left\|g_n\left(\theta\right)\right\|_2^2\right] + \epsilon \Rightarrow$$

$$\sup_{\theta\in\Omega_\theta}\left|\frac{1}{N}\sum_{n=1}^{N}\left\|g_n\left(\theta\right)\right\|_2^2 - \mathbb{E}\left[\left\|g_n\left(\theta\right)\right\|_2^2\right]\right| > \epsilon,$$

we have

$$\begin{aligned} &P\left(\sup_{\theta\in\Omega_\theta}\left|\frac{1}{N}\sum_{n=1}^{N}\left\|g_n\left(\theta\right)\right\|_2^2\right| \geq Q_g^2 + \epsilon\right) \leq \\ &\quad P\left(\sup_{\theta\in\Omega_\theta}\left|\frac{1}{N}\sum_{n=1}^{N}\left\|g_n\left(\theta\right)\right\|_2^2 - \mathbb{E}\left[\left\|g_n\left(\theta\right)\right\|_2^2\right]\right| \leq \epsilon\right), \end{aligned}$$

so

$$P\left(\sup_{\theta\in\Omega_\theta}\left|\frac{1}{N}\sum_{n=1}^{N}\left\|g_n\left(\theta\right)\right\|_2^2\right|\geq C_g^2\right)\xrightarrow[N\to\infty]{}0.$$

An analogous argument holds for $\frac{1}{N}\left\|h_n\left(\theta\right)\right\|_2^2$. Consequently, $P\left(\text{Assumption 3 holds}\right)\xrightarrow[N\to\infty]{}1$.

We now consider Assumption 2. Again, by Keener [2011] Theorem 9.2 applied to each element of the matrix $h_n\left(\theta\right)$, using a union bound over each of the D^2 entries,

$$\sup_{\theta\in\Omega_\theta}\left\|\frac{1}{N}\sum_{n=1}^{N}h_n\left(\theta\right)-\mathbb{E}\left[h_n\left(\theta\right)\right]\right\|_1\xrightarrow[N\to\infty]{p}0.$$

By the converse of Proposition 2, because $\left\|\mathbb{E}\left[h_n\left(\theta\right)\right]^{-1}\right\|_{op}\leq Q_{op}$,

$$\left\|\left(\frac{1}{N}\sum_{n=1}^{N}h_n\left(\theta\right)\right)^{-1}\right\|_{op}>2Q_{op}=C_{op}\Rightarrow$$
$$\left\|\frac{1}{N}\sum_{n=1}^{N}h_n\left(\theta\right)-\mathbb{E}\left[h_n\left(\theta\right)\right]\right\|_1>\frac{1}{2}Q_{op}^{-1}.$$

Consequently,

$$P\left(\left\|\left(\frac{1}{N}\sum_{n=1}^{N}h_n\left(\theta\right)\right)^{-1}\right\|_{op}\geq C_{op}\right)\leq$$
$$P\left(\left\|\frac{1}{N}\sum_{n=1}^{N}h_n\left(\theta\right)-\mathbb{E}\left[h_n\left(\theta\right)\right]\right\|_1\right)\xrightarrow[N\to\infty]{p}0,$$

and $P\left(\text{Assumption 2 holds}\right)\xrightarrow[N\to\infty]{}1$.

Finally, applying Lemma 10 to $\frac{1}{\sqrt{N}}\left\|h\left(\theta_2\right)\right\|_2$,

$$\begin{aligned}\left|\frac{1}{\sqrt{N}}\left\|h\left(\theta_1\right)\right\|_2-\frac{1}{\sqrt{N}}\left\|h\left(\theta_2\right)\right\|_2\right|&\leq\sup_{\theta\in\Omega_\theta}\left\|\frac{\partial}{\partial\theta}\frac{1}{\sqrt{N}}\left\|h\left(\theta\right)\right\|_2\right\|_2\left\|\theta_1-\theta_2\right\|_2\\&\leq\frac{D^2}{\sqrt{N}}\sup_{\theta\in\Omega_\theta}\left\|r\left(\theta\right)\right\|_2\left\|\theta_1-\theta_2\right\|_2\\&=D^2\sqrt{\sup_{\theta\in\Omega_\theta}\frac{1}{N}\left\|r\left(\theta\right)\right\|_2^2}\left\|\theta_1-\theta_2\right\|_2.\end{aligned}$$

Consequently,

$$\left| \frac{1}{\sqrt{N}} \|h(\theta_1)\|_2 - \frac{1}{\sqrt{N}} \|h(\theta_2)\|_2 \right| \geq L_h \|\theta_1 - \theta_2\|_2 \Rightarrow$$

$$D^2 \sqrt{\sup_{\theta \in \Omega_\theta} \frac{1}{N} \|r(\theta)\|_2^2} \geq L_h \Rightarrow$$

$$\sup_{\theta \in \Omega_\theta} \frac{1}{N} \|r(\theta)\|_2^2 - \sup_{\theta \in \Omega_\theta} \mathbb{E}\left[\|r_n(\theta)\|_2^2\right] \geq \frac{L_h^2}{D^4} - \sup_{\theta \in \Omega_\theta} \mathbb{E}\left[\|r_n(\theta)\|_2^2\right] \Rightarrow$$

$$\sup_{\theta \in \Omega_\theta} \left| \frac{1}{N} \|r(\theta)\|_2^2 - \mathbb{E}\left[\|r_n(\theta)\|_2^2\right] \right| \geq \frac{L_h^2}{D^4} - \Lambda_h^2 = \epsilon.$$

However, again by Keener [2011] Theorem 9.2,

$$\sup_{\theta \in \Omega_\theta} \left| \frac{1}{N} \|r(\theta)\|_2^2 - \mathbb{E}\left[\|r_n(\theta)\|_2^2\right] \right| \xrightarrow[N \to \infty]{p} 0,$$

so $P(\text{Assumption 4 holds}) \xrightarrow[N \to \infty]{} 1$. □

B Genomics Experiments Details

We demonstrate the Python and R code used to run and analyze the experiments on the genomics data in a sequence of Jupyter notebooks. The output of these notebooks are included below, though they are best viewed in their original notebook form. The notebooks, as well as scripts and instructions for reproducing our analysis in its entirety, can be found in the git repository rgiordan/AISTATS2019SwissArmyIJ.

fit_model_and_save

February 21, 2019

1 Genomics experiment details.

We demonstrate the infinitesimal jackknife on a publicly available data set of mice gene expression in Shoemaker et al. [2015].

Mice were infected with influenza virus, and gene expression was assessed several times after infection, so the observed data consists of expression levels y_{gt} for genes $g = 1, ..., n_g$ and time points $t = 1, ..., n_t$, where in this case $n_g = 1000$ and $n_t = 42$.

This notebook contains the first of three steps in the analysis. In this notebook, we will first load the data and define a basis with a hyperparameter we wish to select with cross validation. We then describe the two stages of our analysis: a regression stage and a clustering stage. We then save the data for further analysis by the notebooks `load_and_refit` and `calculate_prediction_error`.

This notebook assumes you have already followed the instructions in `README.md` to install the necessary packages and create the dataset.

2 Step 1: Initial fit.

```
In [1]: import matplotlib.pyplot as plt
        %matplotlib inline

        import numpy as np
        import inspect
        import os
        import sys
        import time

        np.random.seed(3452453) # nothing special about this seed (we hope)!
```

```
In [2]: from aistats2019_ij_paper import regression_mixture_lib as rm_lib
        from aistats2019_ij_paper import regression_lib as reg_lib
        from aistats2019_ij_paper import sensitivity_lib as sens_lib
        from aistats2019_ij_paper import spline_bases_lib
        from aistats2019_ij_paper import transform_regression_lib as trans_reg_lib
        from aistats2019_ij_paper import loading_data_utils
        from aistats2019_ij_paper import saving_gmm_utils
        from aistats2019_ij_paper import mse_utils

        import plot_utils_lib
```

2.1 The first stage: Regression

2.1.1 Load data

```
In [3]: # Set bnp_data_repo to be the location of a clone of the repo
        # https://github.com/NelleV/genomic_time_series_bnp
        bnp_data_repo = '../../genomic_time_series_bnp'
        y_train, y_test, train_indx, timepoints = loading_data_utils.load_genomics_data(
            bnp_data_repo,
            split_test_train = True,
            train_indx_file = '../fits/train_indx.npy')
```

```
Loading data from:  ../../genomic_time_series_bnp/data/shoemaker2015reprocessed
```

```
In [4]: n_train = np.shape(y_train)[0]
        print('number of genes in training set: \n', n_train)

        n_test = np.shape(y_test)[0]
        print('number of genes in test set: \n', n_test)

        n_genes = n_train + n_test

        test_indx = np.setdiff1d(np.arange(n_genes), train_indx)
        gene_indx = np.concatenate((train_indx, test_indx))
```

```
number of genes in training set:
 700
number of genes in test set:
 300
```

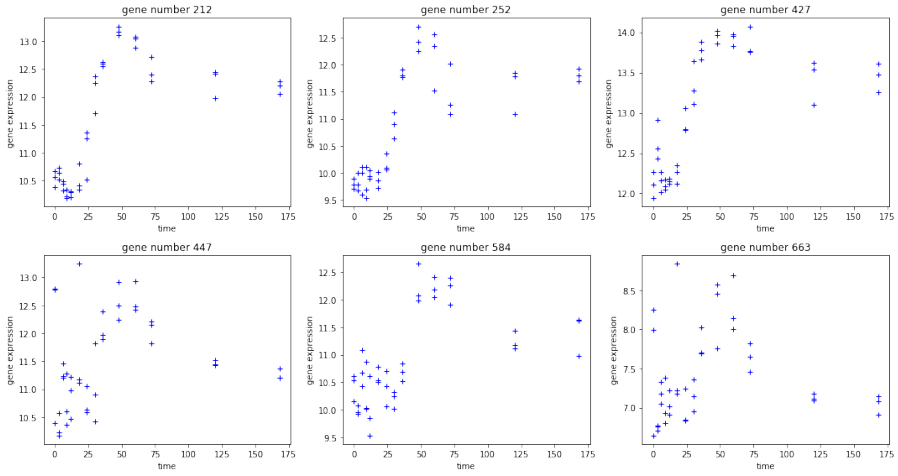
Each gene y_g has 42 observations. Observations are made at 14 timepoints, with 3 replicates at each timepoints.

```
In [5]: n_t = len(timepoints)
        n_t_unique = len(np.unique(timepoints))

        print('timepoints: \n ', timepoints, '\n')
        print('Distinct timepoints: \n', np.sort(np.unique(timepoints)), '\n')
        print('Number of distinct timepoints:', n_t_unique)
```

```
timepoints:
  [0, 0, 0, 3, 3, 3, 6, 6, 6, 9, 9, 9, 12, 12, 12, 18, 18, 18, 24, 24, 24, 30, 30, 30, 36, 36, 3

Distinct timepoints:
 [  0   3   6   9  12  18  24  30  36  48  60  72 120 168]

Number of distinct timepoints: 14
```

Here is the raw data for a few randomly chosen genes.

```
In [6]: f, axarr = plt.subplots(2, 3, figsize=(15,8))

        gene_indx = np.sort(np.random.choice(n_train, 6))

        for i in range(6):
            n = gene_indx[i]
            this_plot = axarr[int(np.floor(i / 3)), i % 3]
            this_plot.plot(timepoints, y_train[n, :].T, '+', color = 'blue');
            this_plot.set_ylabel('gene expression')
            this_plot.set_xlabel('time')
            this_plot.set_title('gene number {}'.format(n))

        f.tight_layout()
```



2.1.2 Define regressors

We model the time course using cubic B-splines. Let α be the degrees of freedom of the B-splines, and this is the parameter we seek to choose using cross-validation.

For a given degrees of freedom, the B-spline basis is given by an $n_t \times n_x$ matrix X_{df}, where the each column of X_{df} is a B-spline basis vector evaluated at the n_t timepoints. Note that n_x increases with increasing degrees of freedom.

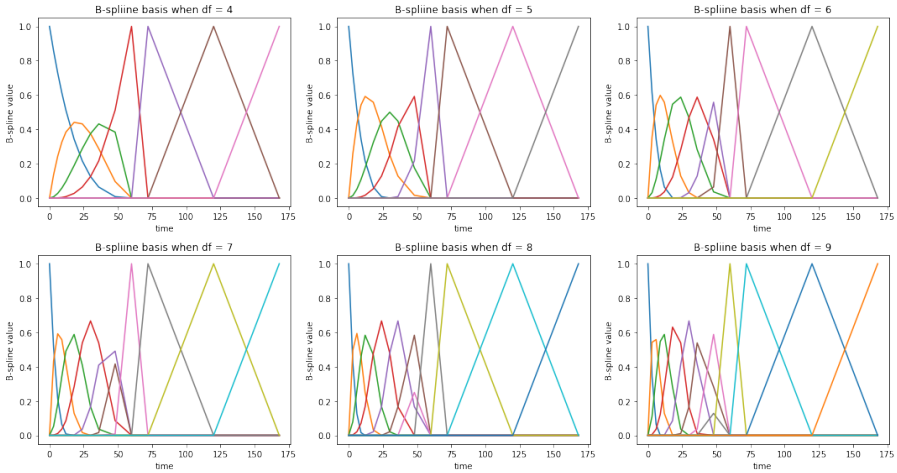
Note that we only use B-splines to smooth the first 11 timepoints. For the last three timepoints, $t = 72, 120, 168$, we use indicator functions on each timepoint as three extra basis vectors. In other words, we append to the regressor matrix three columns, where each column is 1 if $t = 72, 120$, or 168, respectively, and 0 otherwise. We do this to avoid numerical issues in the matrix $X^T X$. Because the later timepoints are more spread out, the B-spline basis are close to zero at the later timepoints, leading to matrices close to being singular.

In [7]:
```
# Simulate passing arguments in on the command line so that the notebook
# looks more like those in ``cluster_scripts``.
class Args():
    def __init__(self):
        pass

args = Args()
args.df = 7
args.degree = 3
args.num_components = 10
```

In [8]:
```
regressors = spline_bases_lib.get_genomics_spline_basis(
    timepoints, df=args.df, degree=3)

regs = reg_lib.Regressions(y_train, regressors)
```

We plot the B-spline matrix for several degrees of freedom below:

In [9]:
```
f, axarr = plt.subplots(2, 3, figsize=(15,8))

i = 0
for df in [4, 5, 6, 7, 8, 9]:
    _regressors = spline_bases_lib.get_genomics_spline_basis(
            timepoints, exclude_num=3, df=df)

    this_plot = axarr[int(np.floor(i / 3)), i % 3]
    this_plot.plot(timepoints, _regressors);
    this_plot.set_xlabel('time')
    this_plot.set_ylabel('B-spline value')
    this_plot.set_title('B-spliine basis when df = {}'.format(df))

    i += 1

f.tight_layout()
```



We display the regressor matrix below.

```
In [10]: plt.matshow(regs.x.T)
         plt.ylabel('basis')
         plt.xlabel('timepoint and replicate')
         plt.title('The (transposed) regressor matrix when df = {}\n'.format(args.df));
```

With the regressor X defined above, for each gene g we model $P\left(y_g|\beta_g,\sigma_g^2\right) = \mathcal{N}\left(y_g|X\beta_g,\sigma_g^2\right)$. In the second stage, we will want to cluster β_g taking into account its uncertainty on each gene. To do this, we wish to estimate the posterior mean $\mathbb{E}[\beta_g|y_g]$ and covariance $\mathrm{Cov}(\beta_g|y_g)$ with flat priors for both β_g and σ_g^2.

For each gene, we estimate the posterior with a mean field variational Bayes (MFVB) approximation $q\left(\sigma_g^2,\beta_g;\hat{\eta}_g\right)$ to the posterior $P\left(\beta_g,\sigma_g^2|y_g\right)$.

In particular, we take $q\left(\sigma_g^2,\beta_g;\hat{\eta}_g\right) = q^*\left(\sigma_g^2\right)q^*\left(\beta_g\right)$, where $q^*\left(\sigma_g^2\right)$ is a dirac delta function, and we optimize over its a location parameter; $q^*\left(\beta_g\right)$ is a Gaussian density and we optimize over its mean and covariance.

The optimal variational approximation has a closed form that is formally identical to the standard frequentist mean and covariance estimate for linear regression. Explicitly, the optimal variational distribution is,

$$q^*(\beta_g) = \mathcal{N}\Big(\beta_g \;\Big|\; (X^TX)^{-1}X^Ty_g,\ \hat{\tau}_g(X^TX)^{-1}\Big)$$
$$q^*(\sigma_g^2) = \delta\{\sigma_g^2 = \hat{\tau}_g\}$$

where $\hat{\tau}_g = \frac{1}{n_t - n_x}\|y_g - X(X^TX)^{-1}X^Ty_g\|_2^2$.

The advantage of the MVFB construction is that $\hat{\eta}_g$ for $g = 1, ..., n_g$ satisfies set of n_g independent M-estimation objectives, allowing us to apply our infinitesimal jackknife results. Specifically, defining $\theta_{reg} := (\eta_1, ..., \eta_{n_g})$, we wish to minimize

$$\begin{aligned} F_{reg}(\theta_{reg}, \alpha) &= \sum_{g=1}^{n_g} KL\left(q\left(\sigma_g^2, \beta_g; \eta_g\right) || P\left(\beta_g, \sigma_g^2 | y_g\right)\right) \\ &= -\sum_{g=1}^{n_g} \mathbb{E}_q\left[\log P\left(\beta_g, \sigma_g^2 | y_g\right)\right] + \mathbb{E}_q\left[\log q\left(\beta_g, \sigma_g^2 | \eta_g\right)\right] \\ &:= \sum_{g=1}^{n_g} F_{reg,g}(\eta_g, \alpha). \end{aligned}$$

Our M-estimator, then, is

$$\frac{\partial F_{reg}(\theta_{reg}, \alpha)}{\partial \theta_{reg}} = 0.$$

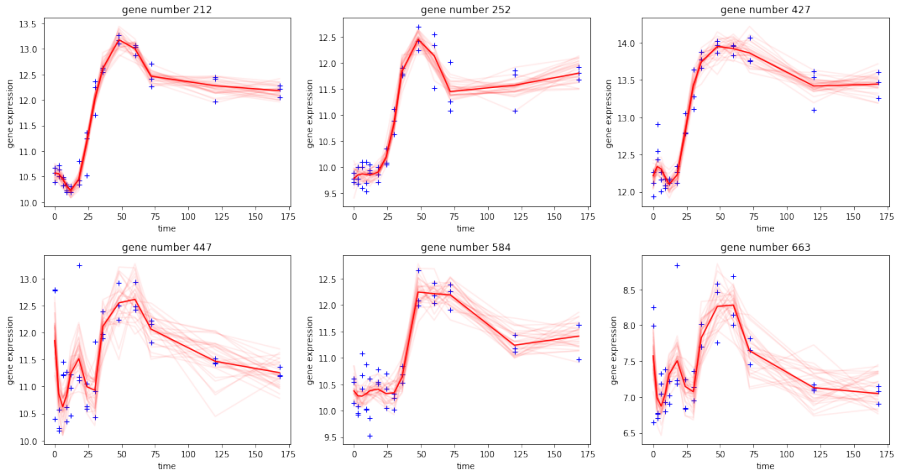
The class `regs` can calculate the optimal variational parameters for each gene. In particular, the variational parameters η_g consist of a variational mean and covariance for β_g, as well as a location estimate for σ_g^2.

```
In [11]: reg_time = time.time()
         opt_reg_params = regs.get_optimal_regression_params()
         reg_time = time.time() - reg_time
         print('Regression time: {} seconds'.format(reg_time))
```

```
Regression time: 0.029132366180419922 seconds
```

Here are what some of the fits look like. Each regression produces a prediction $\hat{y}_g := X\mathbb{E}_q[\beta_g]$, plotted with the heavy red line above. The light red are predictions when β_g is drawn from $q^*(\beta_g)$; the spread of the light red is intended to give a sense of the covariance of β_g.

```
In [12]: f, axarr = plt.subplots(2, 3, figsize=(15,8))

         for i in range(6):
             n = gene_indx[i]
             this_plot = axarr[int(np.floor(i / 3)), i % 3]
             plot_utils_lib.PlotRegressionLine(
                 timepoints, regs, opt_reg_params, n, this_plot=this_plot)

         f.tight_layout()
```



We also define and save data for the test regressions, which we will use later to evaluate out-of-sample performance. The training regressions will be saved below with the rest of the fit.

```
In [13]: regs_test = reg_lib.Regressions(y_test, regressors)
         test_regression_outfile = '../fits/test_regressions.json'
         with open(test_regression_outfile, 'w') as outfile:
             outfile.write(regs_test.to_json())
```

2.2 The second stage: fit a mixture model.

2.2.1 Transform the parameters before clustering

We are interested in the pattern of gene expression, not the absolute level, so we wish to cluster $\hat{y}_g - \bar{\hat{y}}_g$, where $\bar{\hat{y}}_g$ is the average over time points. Noting that the $n_t \times n_t$ matrix $\text{Cov}_q\left(\hat{y}_g - \bar{\hat{y}}_g\right)$ is rank-deficient because we have subtracted the mean, the final step is to rotate $\hat{y}_g - \bar{\hat{y}}_g$ into a basis where the zero eigenvector is a principle axis and then drop that component.

Call these transformed regression coefficients γ_g and observe that $\text{Cov}_q\left(\gamma_g\right)$ has a closed form and is full-rank. It is these γ_gs that we will cluster in the second stage.

We briefly note that the re-centering operation could have been equivalently achieved by making a constant one of the regressors. We chose this implementation because it also allows the user to cluster more complex, non-linear transformations of the regression coefficients, though we leave this extension for future work.

We note that the transformations described in this section are done automatically in the `GMM` class. We are only calculating these transformations here for exposition.

```
In [14]: # Get the matrix that does the transformation.
         transform_mat, unrotate_transform_mat = \
             trans_reg_lib.get_reversible_predict_and_demean_matrix(regs.x)
         trans_obs_dim = transform_mat.shape[0]
```

If T is the matrix that effects the transformation, then

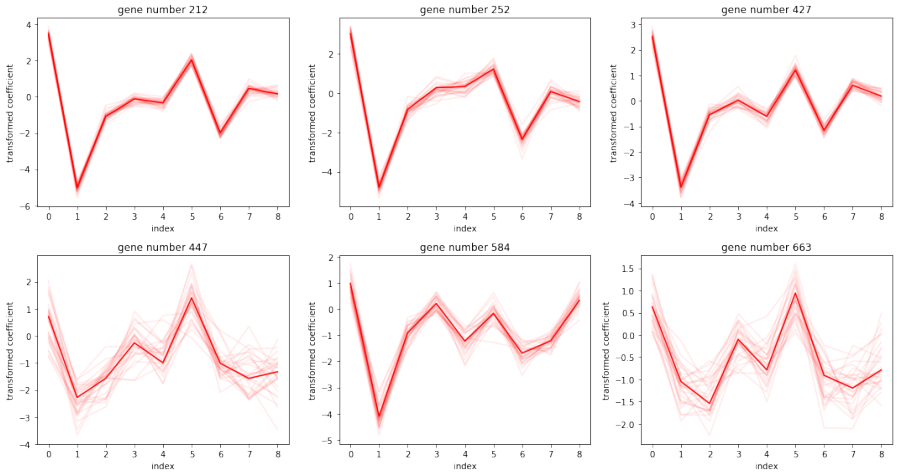
$$\mathbb{E}_q[\gamma_g] = T\mathbb{E}_q[\beta_g]$$
$$\mathrm{Cov}_q(\gamma_q) = T\mathrm{Cov}_q(\beta_g)T^T$$

The transformed parameters are also regression parameters, just in a different space.

In [15]:
```
# Apply the transformation
transformed_reg_params = \
    trans_reg_lib.multiply_regression_by_matrix(
        opt_reg_params, transform_mat)
```

We now visualize the transformed coefficients and their uncertainty.

In [16]:
```
f, axarr = plt.subplots(2, 3, figsize=(15,8))

transformed_beta = transformed_reg_params['beta_mean']
transformed_beta_info = transformed_reg_params['beta_info']

for i in range(6):
    n = gene_indx[i]
    this_plot = axarr[int(np.floor(i / 3)), i % 3]
    this_plot.plot(transformed_beta[n, :], color = 'red');
    this_plot.set_ylabel('transformed coefficient')
    this_plot.set_xlabel('index')
    this_plot.set_title('gene number {}'.format(n))

    # draw from the variational distribution, to plot uncertainties
    for j in range(30):
        transformed_beta_draw = np.random.multivariate_normal(
            transformed_beta[n, :], \
             np.linalg.inv(transformed_beta_info[n]))

        axarr[int(np.floor(i / 3)), i % 3].plot(transformed_beta_draw,
                                                color = 'red', alpha = 0.08);

f.tight_layout()
```



The heavy red lines are the means of the transformed regression coefficients; shaded lines are draws from the variational distribution.

It is these transformed coefficients, γ_g, that we cluster in the second stage.

2.2.2 Estimate an optimal clustering.

We now define a clustering problem for the γ_g. Let n_k be the number of clusters, and $\mu_1, ..., \mu_{n_k}$ be the cluster centers. Also let z_{gk}be the binary indicator for the gth gene belonging to cluster k. We then define the following generative model

$$\begin{aligned}
P(\pi) &= Dirichlet(\omega) \\
P(\mu_k) &= \mathcal{N}(\mu_k|0, \Sigma_0) \quad \text{for} \quad k = 1, ..., n_k \\
P(z_{gk} = 1|\pi_k) &= \pi_k \quad \text{for} \quad k = 1, ..., n_k;\ n = 1, ..., n_g \\
P(\gamma_g|z_{gk} = 1, \mu_k, \eta_g) &= \mathcal{N}(\gamma_g|\mu_k, \text{Cov}_q(\gamma_g) + \epsilon I_{n_t - 1}) \quad \text{for} \quad k = 1, ..., n_k;\ n = 1, ..., n_g.
\end{aligned}$$

where ϵ is a small regularization parameter, which helped our optimization produce more stable results.

We will estimate the clustering using the maximum a posteriori (MAP) estimator of $\theta_{clust} := (\mu, \pi)$. This defines an optimization objective that we seek to minimize:

$$F_{clust}(\theta_{clust}, \theta_{reg}) = -\sum_{g=1}^{n_g} E_{q_z^*}\Big\{\log P(\gamma_g|\eta_g, \mu, \pi, z_g) - \log P(z_g|\pi)\Big\} - \log P(\mu) - \log P(\pi)$$

which, for every value of θ_{reg}, we expect to satisfy

$$\frac{\partial F_{clust}(\theta_{clust}, \theta_{reg})}{\partial \theta_{clust}} = 0.$$

Note that θ_{clust} involves only the ''global" parameters μ and π. We did take a variational distribution for the z_{gk}s, represented by independent Bernoulli distribution, but the optimal q_z^* can be written as a function of μ and π. Hence, our optimization objective only involves these global parameters.

```
In [17]: # Define prior parameters.
         num_components = args.num_components
         epsilon = 0.1
         loc_prior_info_scalar = 1e-5

         trans_obs_dim = regs.x.shape[1] - 1
         prior_params = \
             rm_lib.get_base_prior_params(trans_obs_dim, num_components)
         prior_params['probs_alpha'][:] = 1
         prior_params['centroid_prior_info'] = loc_prior_info_scalar * np.eye(trans_obs_dim)
```

```
In [18]: gmm = rm_lib.GMM(args.num_components,
                          prior_params, regs, opt_reg_params,
                          inflate_coef_cov=None,
                          cov_regularization=epsilon)
```

In our experiment, the number of clusters n_k was chosen to be 10. We set ω to be the ones vector of length n_k. The prior info for the cluster centers Σ_0 is 1e-05$\times I$. ϵ was set to be 0.1.

Let us examine the optimization objective. First, we'll inspect the likelihood terms. What follows is the likelihood given that gene g belongs to cluster k.

```
In [19]: print(inspect.getsource(rm_lib.get_log_lik_nk))
```

```
def get_log_lik_nk(centroids, probs, x, x_infos):
    loc_log_lik = \
        -0.5 * (-2 * np.einsum('ni,kj,nij->nk', x, centroids, x_infos) +
                np.einsum('ki,kj,nij->nk', centroids, centroids, x_infos))

    log_probs = np.log(probs[0, :])
    log_lik_by_nk = loc_log_lik + log_probs.T

    return log_lik_by_nk
```

We can then optimize for q_z^*, which can be parametrized by its mean $\mathbb{E}_{q_z^*}[z]$. We note that this update has a closed form given θ_{clust}, so there is no need to solve an optimization problem to find $q_z^*(z)$. We additionally note that we do not use the EM algorithm, which we found to have exhibit extremely poor convergence rates. Rather, we set $q_z^*(z)$ to its optimal value given θ_{clust} and return the objective as a function of θ_{clust} alone, allowing the use of more general and higher-quality optimization routines.

```
In [20]: print(inspect.getsource(rm_lib.get_e_z))
```

```
def get_e_z(log_lik_by_nk):
    log_const = paragami.simplex_patterns.logsumexp(log_lik_by_nk, axis=1)
    e_z = np.exp(log_lik_by_nk - log_const)
    return e_z
```

With the optimal parameters for z_{nk}, we combine the likelihood term with the prior and entropy terms.

```
In [21]: print(inspect.getsource(rm_lib.wrap_get_loglik_terms))
         print(inspect.getsource(rm_lib.wrap_get_kl))
```

```
def wrap_get_loglik_terms(gmm_params, transformed_reg_params):
    log_lik_by_nk = get_log_lik_nk(
        centroids=gmm_params['centroids'],
        probs=gmm_params['probs'],
        x=transformed_reg_params['beta_mean'],
        x_infos=transformed_reg_params['beta_info'])

    e_z = get_e_z(log_lik_by_nk)

    return log_lik_by_nk, e_z

def wrap_get_kl(gmm_params, transformed_reg_params, prior_params):
    log_lik_by_nk, e_z = \
        wrap_get_loglik_terms(gmm_params, transformed_reg_params)
    log_prior = get_log_prior(
        gmm_params['centroids'], gmm_params['probs'], prior_params)
    return get_kl(log_lik_by_nk, e_z, log_prior)
```

This objective function is wrapped in the `GMM` class method `get_params_kl`.

```
In [22]: print(inspect.getsource(gmm.get_params_kl))
```

```
    def get_params_kl(self, gmm_params):
        """Get the optimization objective as a function of the mixture
        parameters.
        """
        return wrap_get_kl(
            gmm_params, self.transformed_reg_params, self.prior_params)
```

2.2.3 Optimization

For optimization we make extensive use of the autograd library for automatic differentiation and the paragami library for parameter packing and sparse Hessians. These packages' details are beyond the scope of the current notebook.

First, we do a k-means initialization.

```
In [23]: print('Running k-means init.')
         init_gmm_params = \
             rm_lib.kmeans_init(gmm.transformed_reg_params,
                                gmm.num_components, 50)
         print('Done.')
         init_x = gmm.gmm_params_pattern.flatten(init_gmm_params, free=True)
```

```
Running k-means init.
Done.
```

We note that the match between "exact" cross-validation (removing time points and re-optimizing) and the IJ was considerably improved by using a high-quality second-order optimization method. In particular, for these experiments, we employed the Newton conjugate-gradient trust region method (Chapter 7.1 of Wright et al [1999]) as implemented by the method `trust-ncg` in `scipy.optimize`, preconditioned by the Cholesky decomposition of an inverse Hessian calculated at an initial approximate optimum.

We found that first-order or quasi-Newton methods (such as BFGS) often got stuck or terminated at points with fairly large gradients. At such points our method does not apply in theory nor, we found, very well in practice.

The inverse Hessian used for the preconditioner was with respect to the clustering parameters only and so could be calculated quickly, in contrast to the H_1 matrix used for the IJ, which includes the regression parameters as well.

First, run with a low tolerance to get a point at which to evaluate an initial preconditioner.

```
In [24]: gmm.conditioned_obj.reset() # Reset the logging and iteration count.
         gmm.conditioned_obj.set_print_every(1)

         opt_time = time.time()
         gmm_opt, init_x2 = gmm.optimize(init_x, gtol=1e-2)
         opt_time = time.time() - opt_time
```

```
Iter 0: f = -159.11834165
Iter 1: f = -159.67926278
Iter 2: f = -159.97782885
Iter 3: f = -160.15878320
Iter 4: f = -159.59447036
Iter 5: f = -160.19209687
Iter 6: f = -160.27259154
Iter 7: f = -160.29486553
Iter 8: f = -160.33460656
Iter 9: f = -160.34154288
Iter 10: f = -160.32382096
Iter 11: f = -160.34447865
Iter 12: f = -160.34634639
Iter 13: f = -160.34692896
```

Next, set the preconditioner using the square root inverse Hessian at the point `init_x2`.

```
In [25]: tic = time.time()
         h_cond = gmm.update_preconditioner(init_x2)
         opt_time += time.time() - tic
```

The method `optimize_fully` repeats this process of optimizing and re-calculating the preconditioner until the optimal point does not change.

```
In [26]: gmm.conditioned_obj.reset()
         tic = time.time()
         gmm_opt, gmm_opt_x = gmm.optimize_fully(
             init_x2, verbose=True)
         opt_time += time.time() - tic
         print('Optimization time: {} seconds'.format(opt_time))
```

```
Preconditioned iteration 1
  Running preconditioned optimization.
Iter 0: f = -160.34692896
Iter 1: f = -160.34694250
Iter 2: f = -160.34694250
Preconditioned iteration 2
  Getting Hessian and preconditioner.
  Running preconditioned optimization.
Iter 3: f = -160.34694250
Iter 4: f = -160.34694250
Converged.
Optimization time: 8.438910484313965 seconds
```

`paragami` patterns allow conversion between unconstrained vectors and dictionaries of parameter values. After "folding" the optimal `gmm_opt_x`, `opt_gmm_params` contains a dictionary of optimal cluster centroids and cluster probabilities.

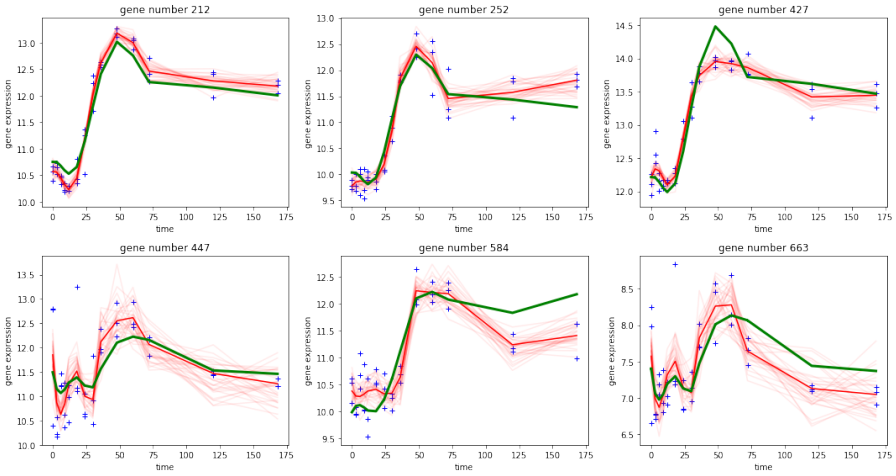
```
In [27]: opt_gmm_params = gmm.gmm_params_pattern.fold(gmm_opt_x, free=True)
         print(opt_gmm_params.keys())
         print(np.sort(opt_gmm_params['probs']))
```

```
odict_keys(['centroids', 'probs'])
[[0.01567608 0.04016882 0.06955236 0.07427946 0.09373695 0.0947442
  0.09653288 0.12626624 0.15739176 0.23165127]]
```

Each gene's regression line has an inferred cluster membership given by $\mathbb{E}_{q_z^*}[z_g]$, and an expected posterior centroid given by $\sum_k \mathbb{E}_{q_z^*}[z_{gk}]\mu_k$. This expected posterior centroid can be untransformed to give a prediction for the observation.

It is the difference between this prediction line — which is a function of the clustering — and the actual data that we consider to be the "error" of the model.

```
In [28]: gmm_pred = mse_utils.get_predictions(gmm, opt_gmm_params, opt_reg_params)

         f, axarr = plt.subplots(2, 3, figsize=(15,8))

         for i in range(6):
             n = gene_indx[i]
             this_plot = axarr[int(np.floor(i / 3)), i % 3]
             plot_utils_lib.PlotRegressionLine(
                 timepoints, regs, opt_reg_params, n, this_plot=this_plot)
             plot_utils_lib.PlotPredictionLine(
                 timepoints, regs, gmm_pred, n, this_plot=this_plot)

         f.tight_layout()
```



2.2.4 Calculating H_1 for the IJ

We seek to choose the degrees of freedom α for the B-splines using cross-validation. We leave out one or more timepoints, and fit using only the remaining timepoints. We then estimate the test error by predicting the value of the genes at the held out timepoints.

To do this, we define time weights w_t by observing that, for each g, the term $\mathbb{E}_q\left[\log P\left(\beta_g, \sigma_g^2 | y_g\right)\right]$ decomposes into a sum over time points:

$$F_{reg,g}\left(\eta_g, \alpha, w\right) := -\sum_{t=1}^{n_t} w_t \left(-\frac{1}{2}\sigma_g^{-2}\left(y_{g,t} - \left(X\beta_g\right)_t\right)^2 - \frac{1}{2}\log\sigma_g^2\right) + \mathbb{E}_q\left[\log q\left(\beta_g, \sigma_g^2 | \eta_g\right)\right].$$

We naturally define $F_{reg}\left(\theta_{reg}, \alpha, w\right) := \sum_{g=1}^{n_g} F_{reg,g}\left(\eta_g, \alpha, w\right)$.

By defining $\theta = (\theta_{clust}, \theta_{reg})$, we then have an M-estimator

$$G(\theta, w, \alpha) := \begin{pmatrix} \frac{\partial F_{reg}(\theta_{reg}, w, \alpha)}{\partial \theta_{reg}} \\ \frac{\partial F_{clust}(\theta_{clust}, \theta_{reg})}{\partial \theta_{clust}} \end{pmatrix} = 0.$$

We can then apply the IJ to approximate the leaving out of various timepoints.

Note that what we call the "Hessian" for this two-step procedure is not really a Hessian, as it is not symmetric. More precisely, it is the Jacobian of G, or what we defined as H_1 in the text.

Calculating H_1 is the most time-consuming part of the infinitesimal jackknife, since the H_1 matrix is quite large (though sparse). However, once H_1 is computed, calculating each $\theta_{IJ}(w)$ is extremely fast.

H_1 can be computed in blocks:

$$H_1 = \begin{pmatrix} \nabla^2_{\theta_{reg}} F_{reg} & 0 \\ \nabla_{\theta_{reg}} \nabla_{\theta_{clust}} F_{clust} & \nabla^2_{\theta_{clust}} F_{clust} \end{pmatrix}$$

The code refers to $\nabla^2_{\theta_{clust}} F_{clust}$ as the "GMM Hessian". It refers to $\nabla_{\theta_{reg}} \nabla_{\theta_{clust}} F_{clust}$ as the "cross Hessian". And it refers to $\nabla^2_{\theta_{reg}} F_{reg}$ as the "regression Hessian", which itself is block diagonal, with each block an observation. Due to details of the implementation of block sparse Hessians using forward mode automatic differnetiation in the class `vittles.SparseBlockHessian`, the code below confusingly refers to each regression parameter as a "block".

When the `FitDerivatives` class is initialized, it calculates these blocks separately and stacks them into the attribute `full_hess`, which is a sparse matrix representing H_1.

```
In [29]: # Even though $H_1$ is not a Hessian, by force of habit we call the time to
         # compute it ``hess_time``.
         hess_time = time.time()
         fit_derivs = sens_lib.FitDerivatives(
             opt_gmm_params, opt_reg_params,
             gmm.gmm_params_pattern, regs.reg_params_pattern,
             gmm=gmm, regs=regs,
             print_every=10)
         hess_time = time.time() - hess_time
         print('Total hessian time: {} seconds'.format(hess_time))
```

```
Initializing FitDerivatives.
Getting t Jacobian.
Getting full Hessian.
   Getting GMM Hessian...
   GMM Hessian time: 2.1917014122009277
   Getting cross Hessian...
   Cross Hessian time: 34.25235605239868
   Getting regression Hessian...
Block index 0 of 66.
Block index 10 of 66.
Block index 20 of 66.
Block index 30 of 66.
```

```
Block index 40 of 66.
Block index 50 of 66.
Block index 60 of 66.
Done differentiating.
   Regression Hessian time: 121.74362897872925
Done with full Hessian.
Total hessian time: 169.59288716316223 seconds
```

2.2.5 Save results as a compressed file.

The results, including H_1, are now saved. To calculate the exact CV, these results (including the preconditioner) will be loaded and the model will be refit with timepoints left out. To calculate the IJ, the same results will be loaded and H_1 will be used to calculate the IJ.

```
In [30]: extra_metadata = dict()
         extra_metadata['opt_time'] = opt_time
         extra_metadata['reg_time'] = reg_time
         extra_metadata['hess_time'] = hess_time
         extra_metadata['df'] = args.df
         extra_metadata['degree'] = args.degree

         npz_outfile = '../fits/initial_fit.npz'
         saving_gmm_utils.save_initial_optimum(
             npz_outfile,
             gmm=gmm,
             regs=regs,
             timepoints=timepoints,
             fit_derivs=fit_derivs,
             extra_metadata=extra_metadata)
```

2.2.6 Bibliography

J. E. Shoemaker, S. Fukuyama, A. J. Eisfeld, D. Zhao, E. Kawakami, S. Sakabe, T. Maemura, T. Gorai, H. Katsura, Y. Muramoto, S. Watanabe, T. Watanabe, K. Fuji, Y. Matsuoka, H. Kitano, and Y. Kawaoka. An Ultrasensitive Mechanism Regulates Influenza Virus-Induced Inflammation. PLoS Pathogens, 11(6):1–25, 2015

S. Wright and J. Nocedal. Numerical optimization. Springer Science, 35(67-68):7, 1999.

load_and_refit

February 21, 2019

1 Step 2: Refit.

In this notebook, we calculate the parameters used for exact CV by refitting the model initially fit in step one, the notebook `fit_model_and_save`.

For expository purposes this notebook calculates the refit for only one weight vector. To compute exact CV, one would perform the corresponding computation for all leave-k-out weight vectors.

In [1]:
```
from copy import deepcopy
import inspect
import matplotlib.pyplot as plt
%matplotlib inline
import numpy as np
import sys
import time

np.random.seed(3452453)

import paragami

from aistats2019_ij_paper import regression_mixture_lib as rm_lib
from aistats2019_ij_paper import saving_gmm_utils
from aistats2019_ij_paper import mse_utils

import plot_utils_lib
```

In [2]:
```
# Load the initial fit.
# This file was produced by the notebook ``fit_model_and_save``.
initial_fit_infile = '../fits/initial_fit.npz'
full_fit, gmm, regs, metadata = \
    saving_gmm_utils.load_initial_optimum(initial_fit_infile)
timepoints = metadata['timepoints']
```

```
Initializing FitDerivatives.
Using provided t_jac.
Using provided full_hess.
```

First, choose some timepoints to leave out.

```
In [3]: # Simulate passing arguments in on the command line.
        class Args():
            def __init__(self):
                pass

        args = Args()
        args.num_times = 1
        args.which_comb = 1
        args.max_num_timepoints = 7
```

The number of points left out (that is, k) is given by `num_times`, which is 1. The largest timepoint we leave out is given by `max_num_timepoints`, which is 7. Because later timepoints are not affected by the smoothing, there is no reason to leave them out.

There are a certain number of ways to leave k out of 7 timepoints, and `which_comb` chooses one of them in the order given by the function `itertools.combinations`. Of course, when $k = 1$, `which_comb` simply chooses which timepoint to leave out. `mse_utils.get_indexed_combination` maps `which_comb` to particular timepoints in a consistent way.

Full exact CV would run this script for all 7 choose k values of `which_comb`.

Because we have repeated measurements at each timepoint, leaving out a single timepoint will correspond to leaving out multiple row of the observation matrix. Those rows are determined by `mse_utils.get_time_weight`, which also returns a weight vector setting these observations' weights to zero.

```
In [4]: lo_inds = mse_utils.get_indexed_combination(
            num_times=args.num_times, which_comb=args.which_comb,
            max_num_timepoints=args.max_num_timepoints)
        new_time_w, full_lo_inds = mse_utils.get_time_weight(lo_inds, timepoints)

        print('Left out timepoint: {}'.format(lo_inds))
        print('Left out observations: {}'.format(full_lo_inds))
        print('Leave-k-out weights: {}'.format(new_time_w))
```

```
Left out timepoint: [1]
Left out observations: [3 4 5]
Leave-k-out weights: [1 1 1 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1]
```

We now re-optimize with the new weights.

Note that we could either start the optimization at the initial optimum (a "warm start") or do a fresh start from k-means. A fresh start is more time consuming but a more stringent test for the accuracy of the IJ. We calculate both, but report results from the fresh start in the paper. In the notebook `examine_and_save_results`, you can choose to examine either set of results.

Here, for consistency with the paper, we re-initialize with k-means.

```
In [5]: regs.time_w = deepcopy(new_time_w)
        reg_params_w = regs.get_optimal_regression_params()
```

```
        gmm.set_regression_params(reg_params_w)

        init_gmm_params = \
            rm_lib.kmeans_init(gmm.transformed_reg_params,
                               gmm.num_components, 50)
        init_x = gmm.gmm_params_pattern.flatten(init_gmm_params, free=True)

        opt_time = time.time()
        gmm_opt, init_x2 = gmm.optimize(init_x, gtol=1e-2)

        print('\tUpdating preconditioner...')
        kl_hess = gmm.update_preconditioner(init_x2)

        print('\tRunning preconditioned optimization...')
        gmm.conditioned_obj.reset()
        reopt, gmm_params_free_w = gmm.optimize_fully(init_x2, verbose=True)
        print(gmm_opt.message)
        opt_time = time.time() - opt_time

        print('Refit time: {} seconds'.format(opt_time))
```

```
Iter 0: f = -153.38003431
Iter 1: f = -152.49438715
Iter 2: f = -153.69147895
Iter 3: f = -153.83779915
Iter 4: f = -154.02397812
Iter 5: f = -153.41393391
Iter 6: f = -154.10396420
Iter 7: f = -154.14366282
Iter 8: f = -154.14261201
Iter 9: f = -154.16417745
Iter 10: f = -154.18307547
Iter 11: f = -154.20711481
Iter 12: f = -154.22118064
Iter 13: f = -154.27402715
Iter 14: f = -154.28739474
Iter 15: f = -154.33849929
Iter 16: f = -154.03580241
Iter 17: f = -154.35421130
Iter 18: f = -154.36910489
Iter 19: f = -154.36872458
Iter 20: f = -154.37238982
Iter 21: f = -154.37722095
Iter 22: f = -154.38186985
Iter 23: f = -154.38410992
        Updating preconditioner...
        Running preconditioned optimization...
Preconditioned iteration 1
```

```
  Running preconditioned optimization.
Iter 0: f = -154.38410992
Iter 1: f = -154.38423176
Iter 2: f = -154.38584092
Iter 3: f = -154.21889674
Iter 4: f = -154.42200228
Iter 5: f = -154.39603234
Iter 6: f = -154.39957947
Iter 7: f = -154.41374585
Iter 8: f = -154.43397491
Iter 9: f = -154.43484046
Iter 10: f = -154.43484816
Iter 11: f = -154.43484816
Preconditioned iteration 2
  Getting Hessian and preconditioner.
  Running preconditioned optimization.
Iter 12: f = -154.43484816
Iter 13: f = -154.43484816
Converged.
Optimization terminated successfully.
Refit time: 14.35115647315979 seconds
```

We now save the results.

```
In [6]: gmm_params_w = \
            full_fit.comb_params_pattern['mix'].fold(
                gmm_params_free_w, free=True)
        refit_comb_params = {
            'mix': gmm_params_w,
            'reg': reg_params_w }
        refit_comb_params_free = \
            full_fit.comb_params_pattern.flatten(refit_comb_params, free=True)
```

```
In [7]: save_filename = \
            '../fits/refit__num_times{}__which_comb{}.npz'.format(
                    args.num_times, args.which_comb)
        print('Saving to {}'.format(save_filename))
        saving_gmm_utils.save_refit(
            outfile=save_filename,
            comb_params_free=refit_comb_params_free,
            comb_params_pattern=full_fit.comb_params_pattern,
            initial_fit_infile=initial_fit_infile,
            time_w=new_time_w,
            lo_inds=lo_inds,
            full_lo_inds=full_lo_inds)
```

```
Saving to ../fits/refit__num_times1__which_comb1.npz
```

calculate_prediction_errors

February 21, 2019

1 Step 3: Calculate the IJ and prediction errors.

In this notebook, for a single weight vector, we calculate the IJ itself as well as the prediction errors for exact CV and IJ. This notebook uses the output of the notebooks `load_and_refit` and `fit_model_and_save`.

In [1]:
```
import numpy as np
import paragami
import vittles
import scipy as sp
from scipy import sparse
import time

import seaborn as sns
import pandas as pd

import matplotlib.pyplot as plt
%matplotlib inline

np.random.seed(3452453)

from aistats2019_ij_paper import regression_lib as reg_lib
from aistats2019_ij_paper import sensitivity_lib as sens_lib
from aistats2019_ij_paper import saving_gmm_utils
from aistats2019_ij_paper import mse_utils

import plot_utils_lib
```

In [2]:
```
# Simulate passing arguments in on the command line.
class Args():
    def __init__(self):
        pass

args = Args()
args.num_times = 1
args.which_comb = 1
args.max_num_timepoints = 7
```

```
In [3]: ##############################
        # Load the original fit.

        print('Loading original fit.')
        initial_fit_infile = '../fits/initial_fit.npz'
        full_fit, gmm, regs, initial_metadata = \
            saving_gmm_utils.load_initial_optimum(initial_fit_infile)

        opt_comb_params = full_fit.get_comb_params()
```

```
Loading original fit.
Initializing FitDerivatives.
Using provided t_jac.
Using provided full_hess.
```

```
In [4]: ##############################
        # Load the test data

        test_regression_infile = '../fits/test_regressions.json'
        with open(test_regression_infile) as infile:
            regs_test = reg_lib.Regressions.from_json(infile.read())

        ########################################
        # Load a refit as specfified by ``args``.

        refit_filename = \
            '../fits/refit__num_times{}__which_comb{}.npz'.format(
                    args.num_times, args.which_comb)

        comb_params_free_refit, comb_params_pattern_refit, refit_metadata = \
            saving_gmm_utils.load_refit(refit_filename)

        time_w = refit_metadata['time_w']
        lo_inds = refit_metadata['lo_inds']
        full_lo_inds = refit_metadata['full_lo_inds']

        assert(comb_params_pattern_refit == full_fit.comb_params_pattern)
        comb_params_refit = comb_params_pattern_refit.fold(
            comb_params_free_refit, free=True)

        time_w = refit_metadata['time_w']
        lo_inds = refit_metadata['lo_inds']
        full_lo_inds = refit_metadata['full_lo_inds']
```

The objects named `comb_params` refer to both the regression and clustering parameters. The name `free` refers to the unconstrained flat value for the parameters as calculated by `paragami`.

```
In [5]: print('Regression pattern: ',
              comb_params_pattern_refit['reg'])
```

```
        print('Clustering pattern: ',
              comb_params_pattern_refit['mix'])
```

```
Regression pattern:  OrderedDict:
        [beta_mean] = NumericArrayPattern (700, 10) (lb=-inf, ub=inf)
        [beta_info] = PatternArray (700,) of PDMatrix 10x10 (diag_lb = 0.0)
        [y_info] = NumericArrayPattern (700,) (lb=0.0, ub=inf)
Clustering pattern:  OrderedDict:
        [centroids] = NumericArrayPattern (10, 9) (lb=-inf, ub=inf)
        [probs] = SimplexArrayPattern (1,) of 10-d simplices
```

1.0.1 Calculate the infinitesimal jackknife.

The `vittles` package makes it easy to calculate linear approximations to the sensitivity of M-estimators to hyperparameters, of which the IJ is a special case. Here, the `HyperparameterSensitivityLinearApproximation` uses the sparse value of H_1 calculated earlier.

Note that H_1 is factorized during the initialization of `weight_sens`, and that it takes relatively little time.

```
In [6]: # Note that if you don't cast the jacobian to a numpy array from
        # a numpy matrix, the output is a 2d-array, causing confusion later.
        weight_sens = vittles.HyperparameterSensitivityLinearApproximation(
            objective_fun=lambda: 0,
            opt_par_value=full_fit.comb_params_free,
            hyper_par_value=regs.time_w,
            hessian_at_opt=sp.sparse.csc_matrix(full_fit.full_hess),
            cross_hess_at_opt=np.array(full_fit.t_jac.todense()))
```

We now use the `weight_sens` object to approximate the "free" value of the combined parameters at `time_w`. The IJ operates in unconstrained space, so we use `paragami` to fold the unconstrained vector back into a dictionary of parameters.

```
In [7]: # Get the infinitesimal jackknife for the refit weight vector.
        lr_time = time.time()
        comb_params_free_lin = \
            weight_sens.predict_opt_par_from_hyper_par(time_w)
        lr_time = time.time() - lr_time
        print('Infinitesimal jackknife time: {}'.format(lr_time))

        comb_params_lin = full_fit.comb_params_pattern.fold(comb_params_free_lin, free=True)
```

```
Infinitesimal jackknife time: 0.0011603832244873047
```

1.0.2 Calculate various prediction errors.

Recall that the prediction error is the difference between the data and the posterior expected cluster centroid for a particular gene. Let us consider the original optimal clustering parameters,

`opt_comb_params['mix']`. To get the test set error on gene g for these parameters, we need to do the following steps:

1. Run the regression for gene g in the test set
2. Classify the regression, calculating $\mathbb{E}_{q_z^*}[z_g]$. This is a function of the clustering parameters and the regression line for gene g.
3. Calculate the expected posterior cluster centroid for gene g, which is $\mu_g^* = \sum_k \mathbb{E}_{q_z^*}[z_{gk}]\mu_k$.
4. Because the transformation discards the mean information, compare the de-meaned data to the estimated centroid: $error_{gt} = \left(y_{gt} - \frac{1}{T}\sum_{t'=1}^{T} y_{gt'}\right) - \mu_{gt}^*$.

Note that step one could re-run the regression either with the original weights or the new weights. We found that this decision does not matter qualitatively. Here and in the paper, we simply classify the original regression, but the notebook `examine_and_save_results` can produce results for oth the original and re-weighted regressions.

We will examine prediction error on the time points that are left out, that is, for observations in `full_lo_inds`.

```
In [8]: print('Calculating prediction error.')

        # Get the training set error on the full data.
        train_error = mse_utils.get_lo_err_folded(
            opt_comb_params,
            keep_inds=full_lo_inds,
            mse_regs=regs,
            mse_reg_params=opt_comb_params['reg'],
            gmm=gmm)

        ############
        # Original fit.

        # Get the optimal test set regressions.
        reg_params_test = regs_test.get_optimal_regression_params()

        # Get the test error for the original fit.
        orig_test_error = mse_utils.get_lo_err_folded(
            opt_comb_params,
            keep_inds=full_lo_inds,
            mse_regs=regs_test,
            mse_reg_params=reg_params_test,
            gmm=gmm)

        orig_pred = mse_utils.get_predictions(
            gmm, opt_comb_params['mix'], reg_params_test)

        # Get the test error for the CV refit.
        cv_error = mse_utils.get_lo_err_folded(
            comb_params_refit,
            keep_inds=full_lo_inds,
```

```
            mse_regs=regs_test,
            mse_reg_params=reg_params_test,
            gmm=gmm)

        cv_pred = mse_utils.get_predictions(
            gmm, comb_params_refit['mix'], reg_params_test)

        # Get the test error for the IJ approximation.
        ij_error = mse_utils.get_lo_err_folded(
            comb_params_lin,
            keep_inds=full_lo_inds,
            mse_regs=regs_test,
            mse_reg_params=reg_params_test,
            gmm=gmm)

        ij_pred = mse_utils.get_predictions(
            gmm, comb_params_lin['mix'], reg_params_test)

Calculating prediction error.
```

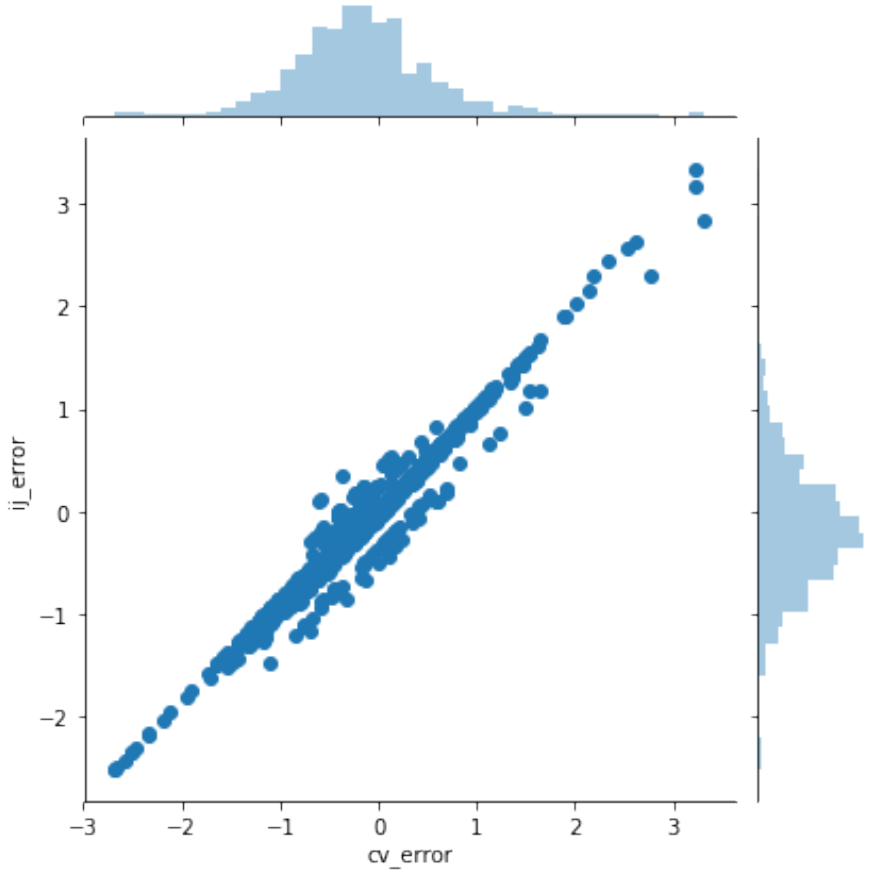
1.0.3 Selected results.

We now make a cursory comparison of the results. For a more detailed analysis, including the results that went into the paper, see the notebook `examine_and_save_results`.

```
In [9]: cv_excess_error = cv_error - orig_test_error
        ij_excess_error = ij_error - orig_test_error

        def GetColDf(col):
            return pd.DataFrame(
                {'cv_error': cv_error[:, col],
                 'cv_excess': cv_excess_error[:, col],
                 'ij_error': ij_error[:, col],
                 'ij_excess': ij_excess_error[:, col],
                 'col': col})

        result = pd.concat([ GetColDf(col) for col in range(len(full_lo_inds)) ])
```

If we simply look at the point-by-point error, CV and IJ are highly correlated.

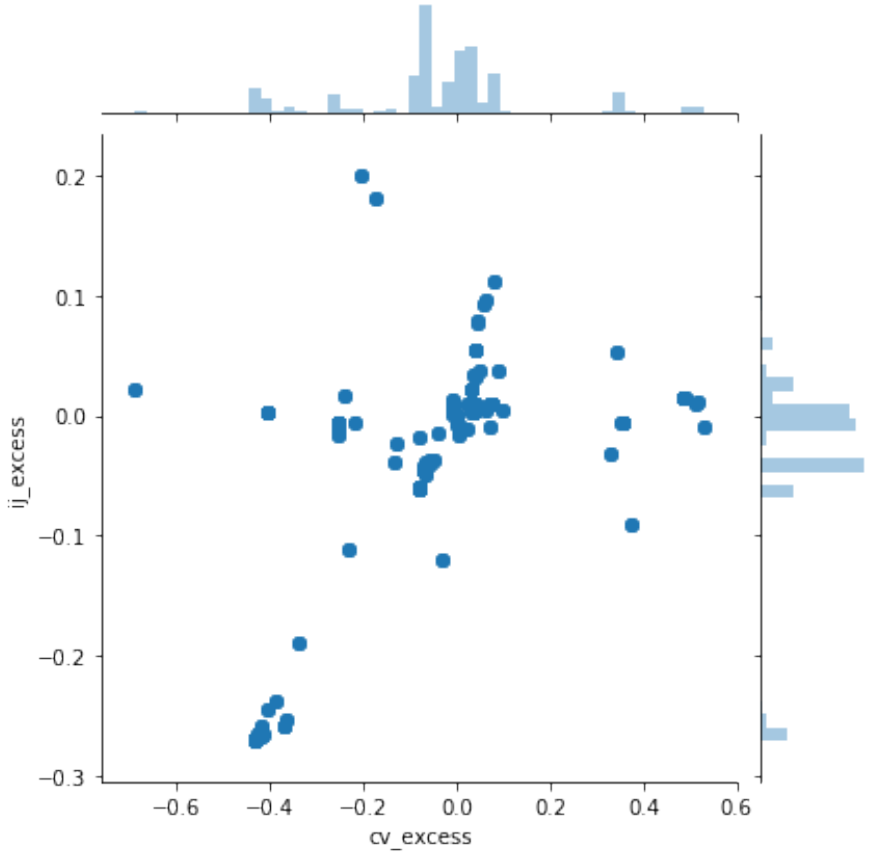
```
In [10]: sns.jointplot(x='cv_error', y='ij_error', data=result);
```



However, this is because the error in each point is dominated by the error at the original optimum. To meaningfully compare the IJ to CV, we should compare the difference between the IJ and CV error and the error at the original optimum. The distribution of these "difference-in-difference" errors is shown in the next plot.

Some clear outliers can be seen. However, note that, in this case, overplotting makes IJ looks worse than it is – in the histograms you can see that most differences are very small.

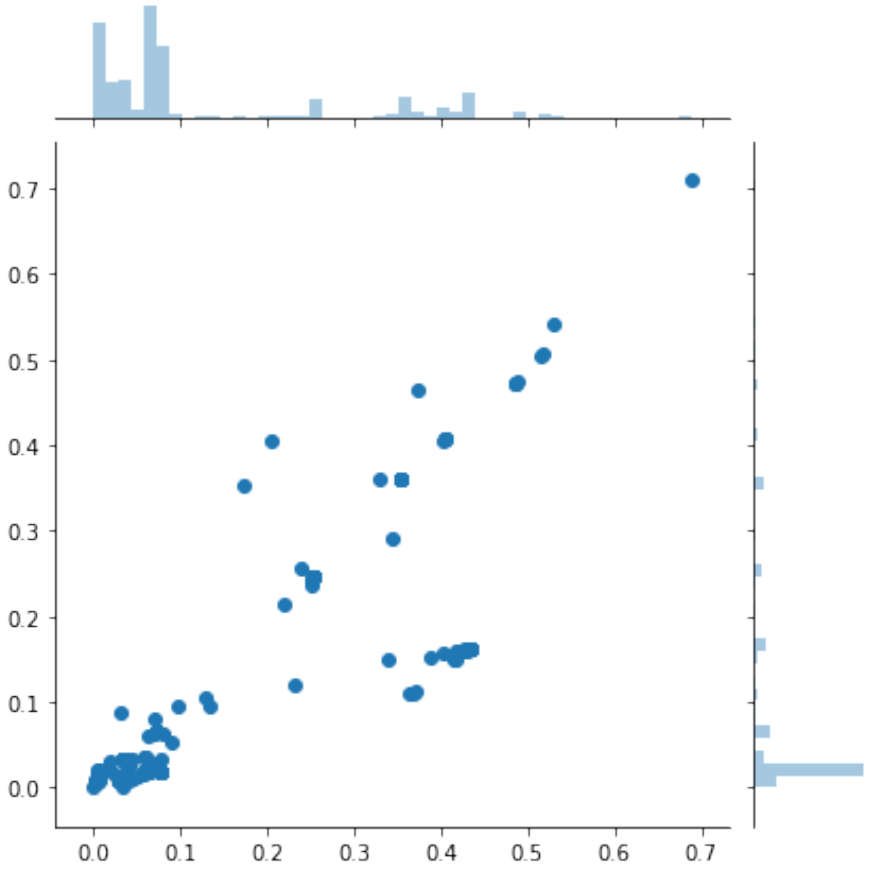
```
In [11]: sns.jointplot(x='cv_excess', y='ij_excess', data=result);
```



As you might expect from a linear approximation, the IJ does the worst when the predicted change for CV is large.

```
In [12]: misfit = np.max(np.abs(cv_excess_error - ij_excess_error), axis=1)
         abs_cv_excess_error = np.max(np.abs(cv_excess_error), axis=1)

         sns.jointplot(abs_cv_excess_error, misfit)
```

```
Out[12]: <seaborn.axisgrid.JointGrid at 0x7f3f74fe1908>
```



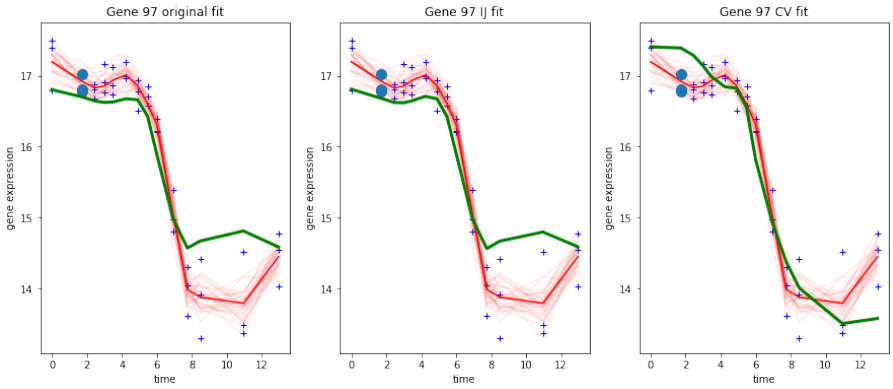
Finally, we visualize some of the genes where IJ badly misestimates the CV error. Clearly, in these cases, re-fitting with the left-out points (shown with large dots) produced large changes that the IJ did not capture. In general, it appears that the IJ errs relative to CV by not moving far enough from the original optimum.

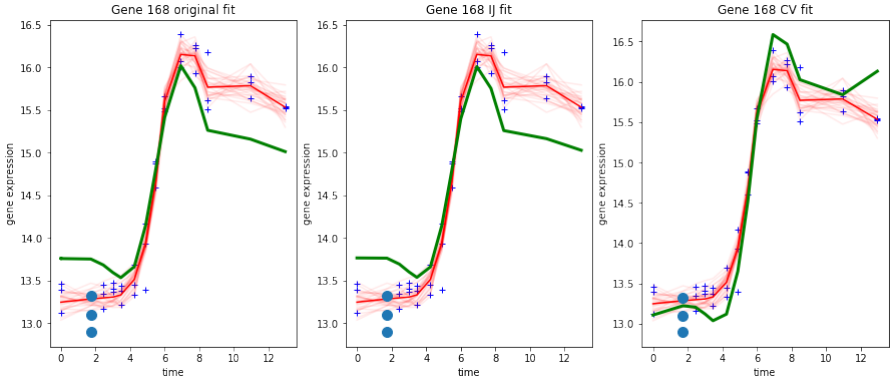
Despite the poor fit on these extreme genes, we stress that most genes exhibited small changes in both CV and IJ. For these genes, IJ performs well enough to capture salient aspects of the estimated out-of-sample error. For more detailed analysis of this point, see the notebook `examine_and_save_results`.

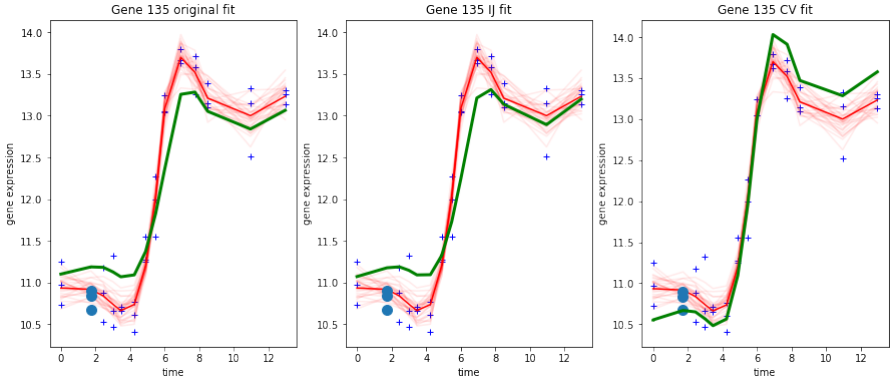
```
In [13]: timepoints = initial_metadata['timepoints']
         timepoints_stretch = np.sqrt(timepoints)

         def PlotGenePredictions(gene_ind):
             _, figs = plt.subplots(1, 3, figsize=(15,6))
```

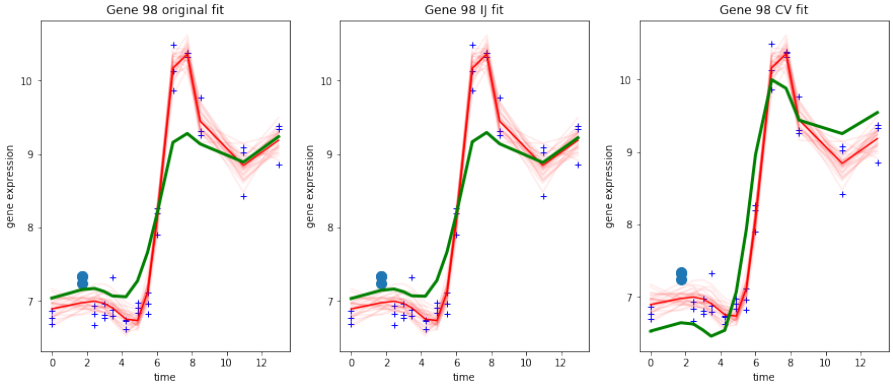
```
    for i in range(3):
        np.random.seed(42)
        plot_utils_lib.PlotRegressionLine(
            timepoints_stretch, regs_test, reg_params_test, gene_ind, this_plot=figs[i]
        figs[i].plot(timepoints_stretch[full_lo_inds],
                     regs_test.y[gene_ind, full_lo_inds], 'o', markersize=10)

    plot_utils_lib.PlotPredictionLine(
        timepoints_stretch, regs_test, orig_pred, gene_ind, this_plot=figs[0])
    figs[0].set_title('Gene {} original fit'.format(gene_ind))

    plot_utils_lib.PlotPredictionLine(
        timepoints_stretch, regs_test, ij_pred, gene_ind, this_plot=figs[1])
    figs[1].set_title('Gene {} IJ fit'.format(gene_ind))

    plot_utils_lib.PlotPredictionLine(
        timepoints_stretch, regs_test, cv_pred, gene_ind, this_plot=figs[2])
    figs[2].set_title('Gene {} CV fit'.format(gene_ind))
```

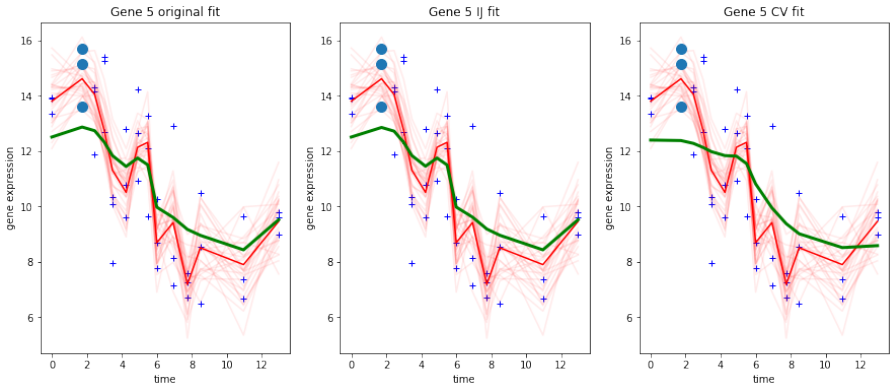
In [14]:
```
worst_fits = np.argsort(-1 * misfit)

for gene in worst_fits[0:5]:
    PlotGenePredictions(gene)
```




Gene 168 original fit
Gene 168 IJ fit
Gene 168 CV fit
gene expression
time


Gene 135 original fit
Gene 135 IJ fit
Gene 135 CV fit
gene expression
time


Gene 98 original fit
Gene 98 IJ fit
Gene 98 CV fit
gene expression
time


Gene 5 original fit
Gene 5 IJ fit
Gene 5 CV fit
gene expression
time

examine_and_save_results

February 21, 2019

1 Detailed analysis of results.

This notebook loads the output of the scripts in the directory `cluster_scripts` (particularly, the final script, `run_slurm_pred_error.py`). It produces the Rdata file that is used for the graphs in the paper as well as a number of supplemental analyses.

In [1]:
```
library(tidyverse)
library(gridExtra)
library(repr) # For setting plot sizes
source("load_python_data_lib.R")
py_main <- InitializePython()
```

```
 Attaching packages  tidyverse 1.2.1
 ggplot2 3.1.0      purrr   0.2.5
 tibble  1.4.2      dplyr   0.7.8
 tidyr   0.8.1      stringr 1.3.1
 readr   1.1.1      forcats 0.3.0
 Conflicts  tidyverse_conflicts()
 dplyr::filter() masks stats::filter()
 dplyr::lag()    masks stats::lag()

Attaching package: gridExtra

The following object is masked from package:dplyr:

    combine


Attaching package: reshape2

The following object is masked from package:tidyr:

    smiths
```

In [2]:
```
# Choose the initialization method.
init_method <- "kmeans" # This is the choice for the paper.
```

```
#init_method <- "warm"

# Choose whether or not to re-run the regressions before calculating test error.
use_rereg <- FALSE # This is the choice for the paper.
#use_rereg <- TRUE

# This is the file that is used in the paper's knitr.
save_dir <- "../../fits"
save_filename <- sprintf("paper_results_init_%s_rereg_%s.Rdata", init_method, use_rereg)
```

1.0.1 Load the saved data for all dfs and k

In [3]:
```
dfs <- list()
metadata_dfs <- list()

for (lo_num_times in 1:3) {
  cat("lo_num_times ", lo_num_times)
  for (df in 4:8) {
    cat(".")
    load_res <- LoadPredictionError(df, lo_num_times, init_method)
    this_refit_err_df <- load_res$refit_err_df
    this_metadata_df <- load_res$metadata_df
    this_refit_err_melt <- MeltErrorColumns(this_refit_err_df)
    dfs[[length(dfs) + 1]] <- this_refit_err_melt
    metadata_dfs[[length(metadata_dfs) + 1]] <- this_metadata_df
  }
  cat("\n")
}
cat("Done.\n")
refit_err_melt <- do.call(bind_rows, dfs)
metadata_df <- do.call(bind_rows, metadata_dfs)
```

```
lo_num_times  1...
lo_num_times  2...
lo_num_times  3...
Done.
```

1.0.2 Metadata (timing, parameter dimensions)

Make a tidy dataframe with the metadata. The parameter length, Hessian time, and initial optimization time are all reported in the text of the paper. Their values will be derived from this dataframe in knitr.

In [4]:
```
metadata_df <-
    metadata_df %>%
    mutate(lr_hess_time=total_lr_time + initial_hess_time,
           avg_lr_time=total_lr_time / num_comb,
           avg_refit_time=total_refit_time / num_comb,
```

```
                param_length=gmm_param_length + reg_param_length)
        print(names(metadata_df))

        select(metadata_df, df, param_length) %>%
            group_by(df) %>%
            summarize(param_length=unique(param_length))

        select(metadata_df, df, initial_hess_time, initial_opt_time) %>%
            group_by(df) %>%
            summarize(initial_hess_time=median(initial_hess_time),
                      initial_opt_time=median(initial_opt_time))

        round(median(metadata_df$initial_opt_time), digits=-1)
```

```
 [1] "num_comb"          "total_lr_time"     "total_refit_time"
 [4] "initial_opt_time"  "initial_reg_time"  "initial_hess_time"
 [7] "gmm_param_length"  "reg_param_length"  "df"
[10] "lo_num_times"      "init_method"       "lr_hess_time"
[13] "avg_lr_time"       "avg_refit_time"    "param_length"
```

df	param_length
4	25325
5	31643
6	38661
7	46379
8	54797

df	initial_hess_time	initial_opt_time
4	275.7295	31.44656
5	295.0325	41.84182
6	359.6855	35.11145
7	478.7345	50.88843
8	584.4987	77.02919

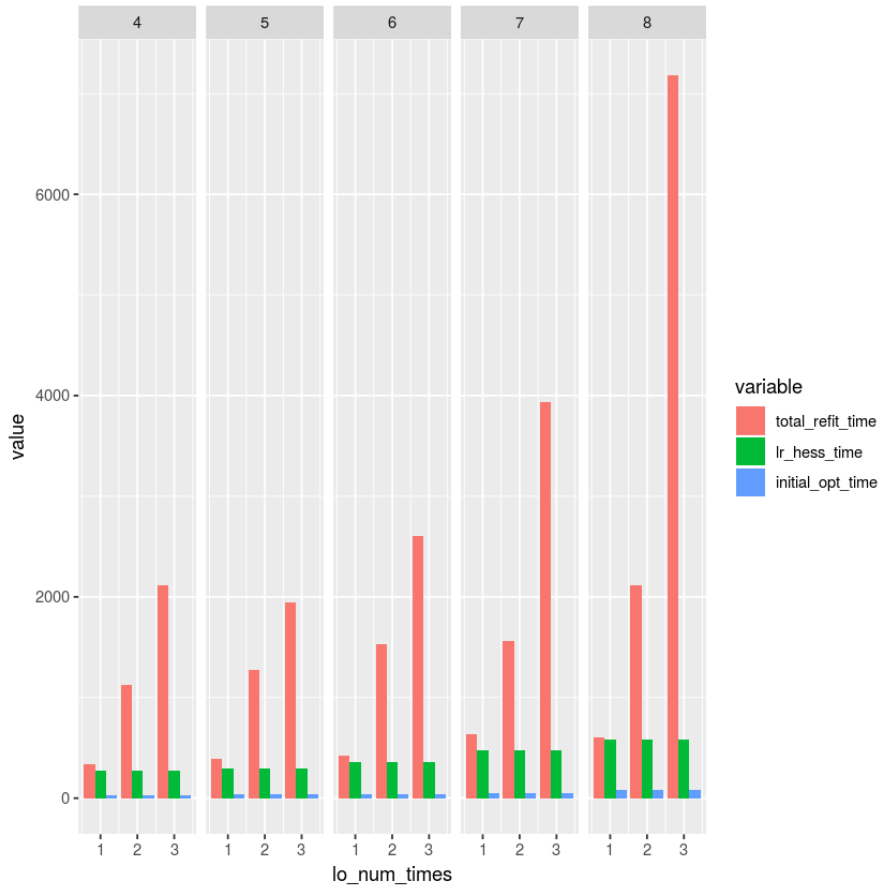
40

Make a dataframe for the timing plot from the metadata.

```
In [5]: metadata_graph_df <-
            metadata_df %>%
            select(df, lo_num_times, total_refit_time, lr_hess_time, initial_opt_time) %>%
            melt(id.vars=c("lo_num_times", "df"))
        head(metadata_graph_df)
```

lo_num_times	df	variable	value
1	4	total_refit_time	338.1638
1	5	total_refit_time	391.6006
1	6	total_refit_time	423.8322
1	7	total_refit_time	632.2635
1	8	total_refit_time	599.0894
2	4	total_refit_time	1123.7316

```
In [6]: ggplot(metadata_graph_df) +
            geom_bar(aes(x=lo_num_times, y=value, fill=variable, group=variable),
                     stat="identity", position=position_dodge()) +
            facet_grid( ~ df)
```



1.0.3 Calculate prediction errors

Make summaries of prediction error for various methods and datasets.

```
In [7]: # In-sample IJ error.
        lr_df <-
          refit_err_melt %>%
          filter(rereg==use_rereg, method=="lin", test==FALSE, measure=="err") %>%
```

```
  rename(error=value) %>%
  mutate(output="lin_in_sample")

# In-sample CV error.
cv_df <-
  refit_err_melt %>%
  filter(rereg==use_rereg, method=="ref", test==FALSE, measure=="err") %>%
  rename(error=value) %>%
  mutate(output="cv_in_sample")

# In-sample training error (no points left out).
train_df <-
  refit_err_melt %>%
  filter(rereg==use_rereg, method=="ref", test==FALSE, measure=="train_err") %>%
  rename(error=value) %>%
  mutate(output="train_error")

# Out-of-sample test error.
test_df <-
  refit_err_melt %>%
  filter(rereg==use_rereg, method=="ref", test==TRUE, measure=="train_err") %>%
  rename(error=value) %>%
  mutate(output="test_error")

refit_for_df_choice <- bind_rows(
  lr_df, cv_df, test_df, train_df)
```

```
In [8]: head(refit_for_df_choice)
```

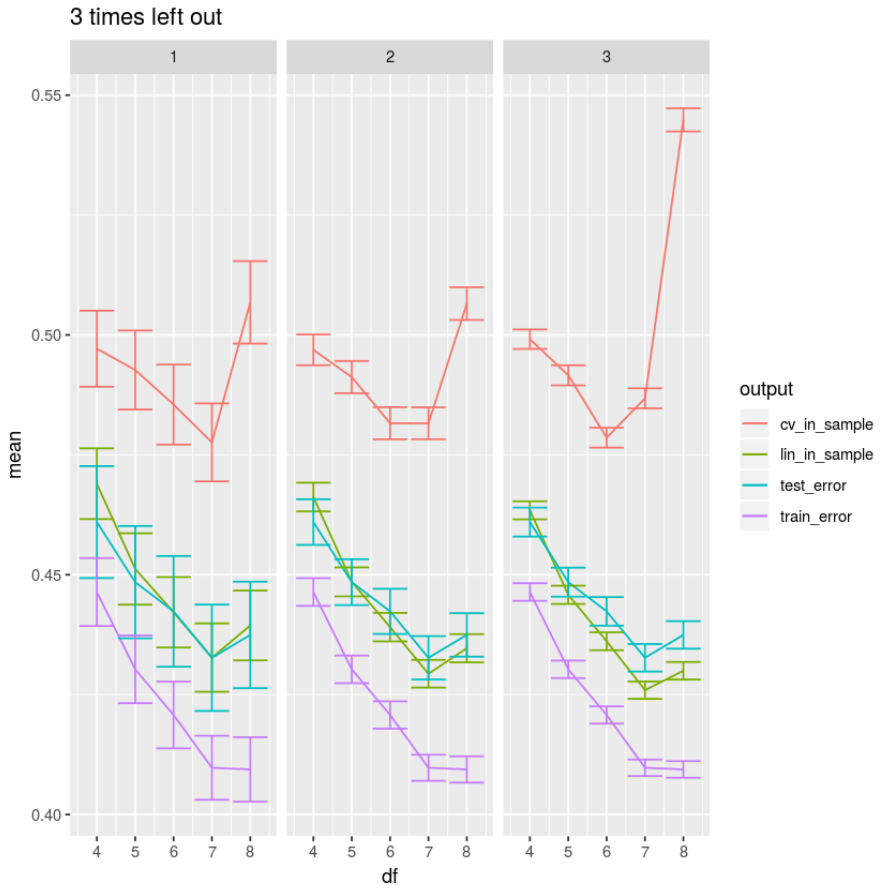
test	method	comb	rereg	gene	df	lo_num_times	time	measure	error	output
FALSE	lin	0	FALSE	0	4	1	0	err	1.0088933	lin_in_sam
FALSE	lin	0	FALSE	1	4	1	0	err	0.1243607	lin_in_sam
FALSE	lin	0	FALSE	2	4	1	0	err	-0.4340983	lin_in_sam
FALSE	lin	0	FALSE	3	4	1	0	err	-0.2203431	lin_in_sam
FALSE	lin	0	FALSE	4	4	1	0	err	1.9032786	lin_in_sam
FALSE	lin	0	FALSE	5	4	1	0	err	-0.2876837	lin_in_sam

Make a tidy dataframe for choosing `df`. The graph in the paper will be based on this dataframe.

Note that most of the signal for choosing `df` is already in the training data error. However, there is an uptick in error in both CV and IJ for `df=8` which is not captured by the training data error.

```
In [9]: refit_err_summary <-
          refit_for_df_choice %>%
          group_by(output, df, lo_num_times) %>%
          mutate(esize=abs(error)) %>%
          summarize(med=median(esize),
                    mean=mean(esize),
                    n_obs=n(),
                    se=sd(esize) / sqrt(n_obs),
```

```
            qlow=quantile(esize, 0.25),
            qhigh=quantile(esize, 0.75))

ggplot(refit_err_summary) +
  geom_line(aes(x=df, y=mean, group=output, color=output)) +
  geom_errorbar(aes(x=df, ymin=mean - 2 * se, ymax=mean + 2 * se,
                    group=output, color=output)) +
  facet_grid(~ lo_num_times) +
  ggtitle(sprintf("%d times left out", lo_num_times))
```



1.0.4 Gene-by-gene accuracy measures.

```
In [10]: refit_err_plot <-
           refit_err_melt %>%
```

```
          filter(rereg==use_rereg) %>%
          dcast(df + lo_num_times + test + comb + rereg + gene + time ~ method + measure,
                value.var=c("value"))
```

We now look at the correlation between the CV and IJ prediction errors across genes. For each `df` and `k`, there are a number of different combinations of left-out points. We report the median, min, and max correlation coefficients across these combinations of left-out points.

First, we show the correlation between the raw prediction errors. Although the correlation is quite high, this is because the training error at the original optimum is the principle source of variation in the errors across genes, and this quantity is common to both CV and IJ.

```
In [11]: err_corr <- refit_err_plot %>%
           filter(test==FALSE, rereg==use_rereg) %>%
           group_by(df, lo_num_times, comb) %>%
           summarize(r=cor(lin_err, ref_err)) %>%
           group_by(df, lo_num_times) %>%
           summarize(med_r=median(r), min_r=min(r), max_r=max(r))

         print("Correlation between error: ")
         print(err_corr)
```

```
[1] "Correlation between error: "
# A tibble: 15 x 5
# Groups:   df [?]
      df lo_num_times med_r min_r max_r
   <int>        <int> <dbl> <dbl> <dbl>
 1     4            1 0.974 0.949 0.984
 2     4            2 0.975 0.902 0.992
 3     4            3 0.967 0.871 0.991
 4     5            1 0.963 0.856 0.983
 5     5            2 0.966 0.860 0.984
 6     5            3 0.947 0.759 0.981
 7     6            1 0.980 0.807 0.985
 8     6            2 0.968 0.835 0.986
 9     6            3 0.929 0.759 0.983
10     7            1 0.962 0.794 0.974
11     7            2 0.952 0.737 0.976
12     7            3 0.914 0.599 0.974
13     8            1 0.962 0.703 0.971
14     8            2 0.941 0.663 0.974
15     8            3 0.829 0.251 0.958
```

A more meaningful measure is the correlation in the excess error for IJ and CV over the error at the original fit.

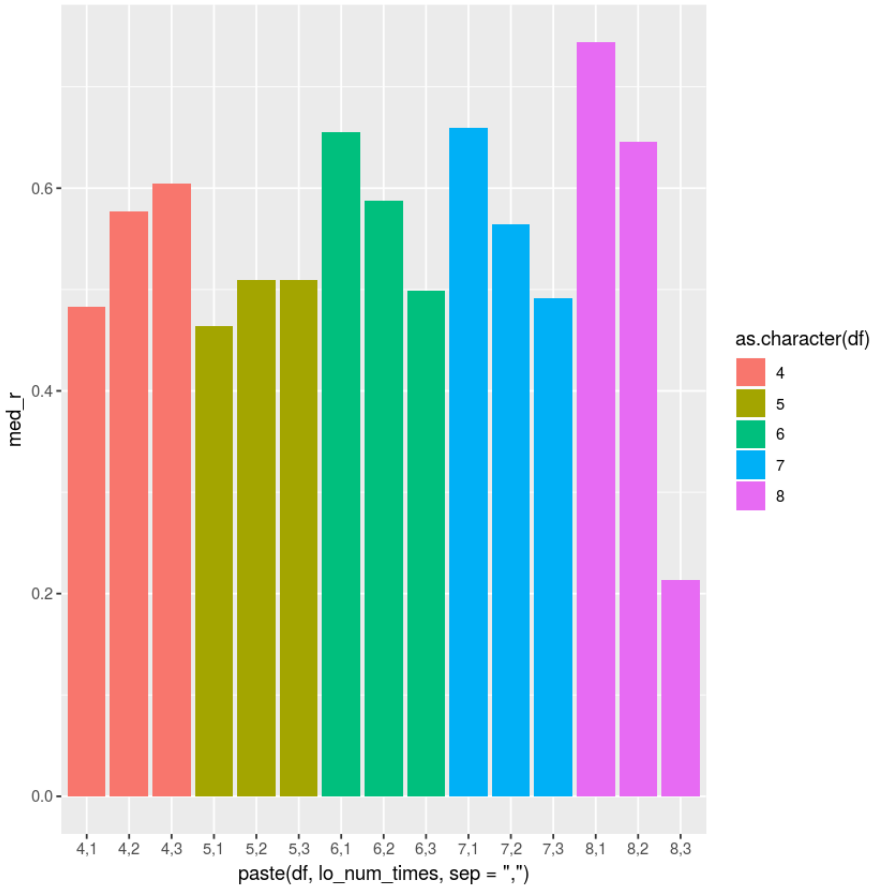
```
In [12]: diff_corr <- refit_err_plot %>%
           filter(test==FALSE, rereg==use_rereg) %>%
           group_by(df, lo_num_times, comb) %>%
```

```
            summarize(r=cor(lin_e_diff, ref_e_diff)) %>%
            group_by(df, lo_num_times) %>%
            summarize(med_r=median(r), min_r=min(r), max_r=max(r))

          print("Correlation between difference from train error: ")
          print(diff_corr)
```

```
[1] "Correlation between difference from train error: "
# A tibble: 15 x 5
# Groups:   df [?]
      df lo_num_times med_r   min_r max_r
   <int>        <int> <dbl>   <dbl> <dbl>
 1     4            1 0.483  0.0956 0.844
 2     4            2 0.577  0.277  0.828
 3     4            3 0.605  0.303  0.833
 4     5            1 0.464  0.143  0.728
 5     5            2 0.510  0.330  0.709
 6     5            3 0.510  0.312  0.671
 7     6            1 0.655  0.368  0.783
 8     6            2 0.588  0.218  0.845
 9     6            3 0.499  0.0701 0.737
10     7            1 0.660  0.512  0.760
11     7            2 0.564  0.224  0.863
12     7            3 0.491  0.0344 0.801
13     8            1 0.744  0.380  0.900
14     8            2 0.646  0.166  0.862
15     8            3 0.214 -0.226  0.767
```

For higher degrees of freedom, increasing the number of left-out points seems to decrease the IJ's accuracy, as you might expect.

```
In [13]: ggplot(diff_corr) +
             geom_bar(aes(x=paste(df, lo_num_times, sep=","),
                          y=med_r, fill=as.character(df)), stat="identity")
```

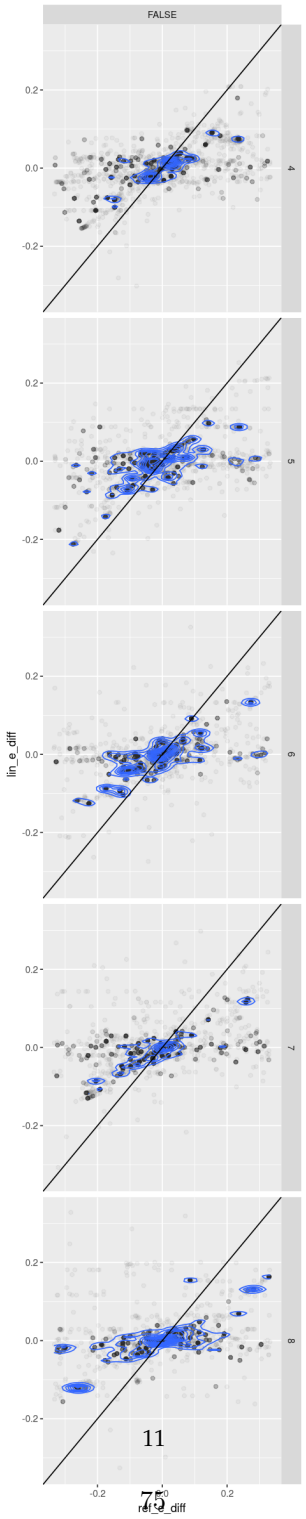


Plot the densities of the IJ and CV with points to show outliers. This is a graphical version of the results summarized by the correlation tables above.

```
In [14]: # There are a few outliers, so limit the extent of the plot so that
         # the bulk of the distribution is visible.
         qlim <- quantile(refit_err_plot$ref_e_diff, c(0.1, 0.9))

         options(repr.plot.width=4, repr.plot.height=20)

         # This plot, or ones like it, is probably the best measure of
         # the accuracy of the IJ.
         ggplot(filter(refit_err_plot, test == FALSE, lo_num_times==1)) +
             geom_point(aes(x=ref_e_diff, y=lin_e_diff), alpha=0.01) +
             geom_density2d(aes(x=ref_e_diff, y=lin_e_diff)) +
```

```
            geom_abline(aes(slope=1, intercept=0)) +
            facet_grid(df ~ rereg) +
            xlim(qlim[1], qlim[2]) + ylim(qlim[1], qlim[2])
```

```
Warning message:
Removed 10770 rows containing non-finite values (stat_density2d).Warning message:
Removed 10770 rows containing missing values (geom_point).
```


FALSE
4
5
6
7
8
0.2
0.0
-0.2
lin_e_diff
ref_e_diff

1.0.5 Save results for plotting in the paper.

In [15]:
```
print(sprintf("Saving to %s", file.path(save_dir, save_filename)))
save(refit_err_summary,
     metadata_df,
     diff_corr,
     err_corr,
     file=file.path(save_dir, save_filename))
```

```
[1] "Saving to ../../fits/paper_results_init_kmeans_rereg_FALSE.Rdata"
```